3D Face Tracking Using Appearance Registration and Robust Iterative Closest Point Algorithm*

Fadi Dornaika and Angel D. Sappa

Computer Vision Center
Edifici O, Campus UAB
08193 Bellaterra, Barcelona, Spain
{dornaika, sappa}@cvc.uab.es

Abstract. Recently, researchers proposed deterministic and statistical appearance-based 3D face tracking methods that can successfully tackle the image variability and drift problems. However, appearance-based methods dedicated to 3D face tracking may suffer from inaccuracies since they are not very sensitive to out-of-plane motion variations. On the other hand, the use of dense 3D facial data provided by a stereo rig or a range sensor can provide very accurate 3D face motions/poses. However, this paradigm requires either an accurate facial feature extraction or a computationally expensive registration technique (e.g., the Iterative Closest Point algorithm). In this paper, we propose a 3D face tracker that is based on appearance registration and on a fast variant of a robust Iterative Closest Point algorithm. The resulting 3D face tracker combines the advantages of both appearance-based trackers and 3D data-based trackers. Experiments on real video data show the feasibility and usefulness of the proposed approach.

1 Introduction

The ability to detect and track human heads and faces in video sequences is useful in a great number of applications, such as human-computer interaction and gesture recognition. There are several commercial products capable of accurate and reliable 3D head position and orientation estimation (e.g., the acoustic tracker system Mouse[1]). These are either based on magnetic sensors or on special markers placed on the face; both practices are encumbering, causing discomfort and limiting natural motion. Vision-based 3D face tracking provides an attractive alternative since vision sensors are not invasive and hence natural motions can be achieved [1]. However, detecting and tracking faces in video sequences is a challenging task.

Recently, deterministic and statistical appearance-based 3D face tracking methods have been proposed and used by some researchers [2,3,4]. These methods can successfully tackle the image variability and drift problems by using

* This work was supported by the MEC project TIN2005-09026 and The Ramón y Cajal Program.

[1] www.vrdepot.com/vrteclg.htm

A. Levi et al. (Eds.): ISCIS 2006, LNCS 4263, pp. 532–541, 2006.

deterministic or statistical models for the global appearance of a special object class: the face. However, appearance-based methods dedicated to full 3D face tracking may suffer from some inaccuracies since these methods are not very sensitive to out-of-plane motion variations. On the other hand, the use of dense 3D facial data provided by a stereo rig or a range sensor can provide very accurate 3D face motions. Unfortunately, computing the 3D face motions from the stream of dense 3D facial data is not straightforward. Indeed, inferring the 3D face motion from the dense 3D data needs an additional process. This process can be the detection of some particular facial features in the range data/images from which the 3D face pose can be inferred. For example, in [5], the 3D nose ridge is detected and then used for computing the 3D face pose. Alternatively, one can perform a registration between 3D data obtained at different time instants in order to infer the relative 3D motions. The most common registration technique is the Iterative Closest Point (ICP) [6]. The ICP algorithm and its variants can provide accurate 3D motions but they are computationally very expensive.

The main contribution of this paper is a robust 3D face tracker that combines the advantages of both appearance-based trackers and 3D data-based trackers. First, the 3D face pose is recovered using an appearance registration technique. Second, possible errors associated with the obtained 3D face pose are reduced using a robust Iterative Closest Point algorithm, which registers a 3D mesh to a 3D facial surface provided by a stereo system. We show that this scheme is extremely faster than a raw implementation of the ICP algorithm.

The remainder of this paper proceeds as follows. Section 2 introduces our deformable 3D facial model. Section 3 summarizes the adaptive appearance-based tracker that tracks in real-time the 3D face pose and some facial actions. It gives some evaluation results. Section 4 describes an improvement step based on a robust ICP algorithm. Section 5 gives some experimental results.

2 Modeling Faces

A deformable 3D model. In our study, we use the 3D face model *Candide* [7]. This 3D deformable wireframe model was first developed for the purpose of model-based image coding and computer animation. The 3D shape of this wireframe model is directly recorded in coordinate form. It is given by the coordinates of the 3D vertices $\mathbf{P}_i, i = 1, \ldots, n$ where n is the number of vertices. Thus, the shape up to a global scale can be fully described by the $3n$-vector $\mathbf{g}$; the concatenation of the 3D coordinates of all vertices $\mathbf{P}_i$. The vector $\mathbf{g}$ is written as:

$$\mathbf{g} = \mathbf{g}_s + \mathbf{A}\,\boldsymbol{\tau_a} \tag{1}$$

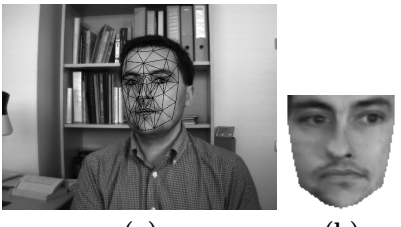
where $\mathbf{g}_s$ is the static shape of the model, $\boldsymbol{\tau_a}$ the animation control vector, and the columns of $\mathbf{A}$ are the Animation Units. In this study, we use six modes for the facial Animation Units (AUs) matrix $\mathbf{A}$. We have chosen the six following AUs: lower lip depressor, lip stretcher, lip corner depressor, upper lip raiser, eyebrow

lowerer, outer eyebrow raiser. These AUs are enough to cover most common facial animations. Moreover, they are essential for conveying emotions.

In equation (1), the 3D shape is expressed in a local coordinate system. However, one should relate the 3D coordinates to the image coordinate system. To this end, we adopt the weak perspective projection model. We neglect the perspective effects since the depth variation of the face can be considered as small compared to its absolute depth. Thus, the state of the 3D wireframe model is given by the 3D face pose parameters (three rotations and three translations) and the internal face animation control vector $\boldsymbol{\tau_a}$. This is given by the 12-dimensional vector $\mathbf{b}$:

$$\mathbf{b} = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \boldsymbol{\tau_a}^T]^T \tag{2}$$

Shape-free facial textures. A face texture is represented as a shape-free texture (geometrically normalized image). The geometry of this image is obtained by projecting the static shape $\mathbf{g}_s$ (neutral shape) using a centered frontal 3D pose onto an image with a given resolution. The texture of this geometrically normalized image is obtained by texture mapping from the triangular 2D mesh in the input image (see figure 1) using a piece-wise affine transform (see [7] for more details).



(a) **(b)**

Fig. 1. (a) an input image with correct adaptation. **(b)** the corresponding shape-free facial image.

3 Tracking Using Adaptive Appearance Registration

Tracking the face and facial actions in a video is carried out by estimating the vector $\mathbf{b}_t$ for every frame. In [8], we have developed an efficient technique for the estimation of the vector $\mathbf{b}$ from the stream of images, by minimizing a distance between the incoming warped frame and the current *shape-free* appearance of the face. This minimization is carried out using a Gauss-Newton method. The statistics of the *shape-free* appearance as well as the gradient matrix are updated every frame. This scheme leads to a fast and robust tracking algorithm. On a 3.2 GHz PC, a non-optimized C code of the approach computes the 3D face pose and the six facial actions in 50 ms. About half that time is required if one is only interested in computing the 3D face pose parameters.

3.1 Accuracy Evaluation

Figure 2 depicts the proposed monocular tracker errors associated with a 300-frame long sequence which contains rotational and translational out-of-plane face motions. This evaluation has been obtained by the joint use of facial surfaces and of the ICP algorithm [9]. The nominal absolute depth of the head was about 65 cm, and the focal length of the camera was 824 pixels. As can be seen, the out-of-plane motion errors can be large for some frames for which there is a room for improvement. Moreover, this evaluation has confirmed the general trend of appearance-based trackers, that is, the out-of-plane motion parameters are more affected by errors than the other parameters.

One expects that the monocular tracker accuracy can be improved if an additional cue is used. In our case, the additional cue will be the 3D data associated to the mesh vertices provided by stereo reconstruction. Although the use of stereo data may seem as an excess requirement, recall that cheap and compact stereo systems are now widely available (e.g., `[www.ptgrey.com]`).

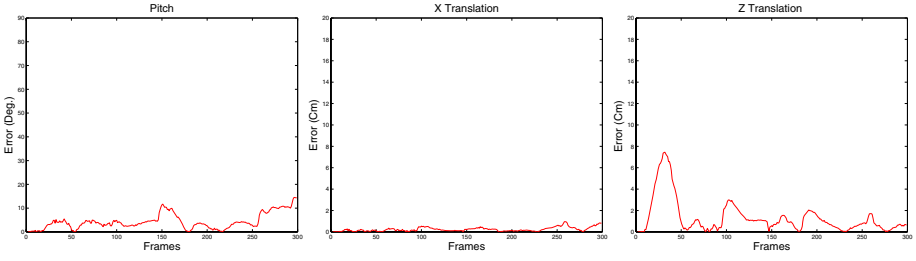


Fig. 2. 3D face pose errors computed by the ICP algorithm

4 Improving the 3D Face Pose Using a Robust ICP Algorithm

In this section, we describe the proposed improvement which is based on a fast and robust Iterative Closest Point algorithm. The ICP algorithm is widely used for geometric alignment of 3D models of rigid objects when an initial estimate of the relative transform is known, or equivalently the two surfaces are roughly aligned. This is consistent with our case since the 3D mesh is roughly aligned with the current 3D facial surface using the estimated appearance-based 3D face

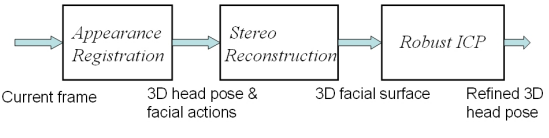


Fig. 3. The main steps of the developed 3D face tracker

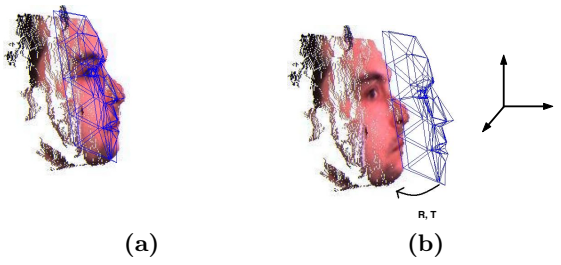


Fig. 4. (a) An ideal case where the appearance-based 3D face pose corresponds to the true 3D face pose. **(b)** A real case where the appearance-based 3D face pose does not exactly correspond to the true 3D face pose. It follows that the improvement is simply the rigid 3D displacement [**R**|**T**] that aligns the two surfaces. This rigid 3D transform can be recovered using a robust Iterative Closest Point algorithm.

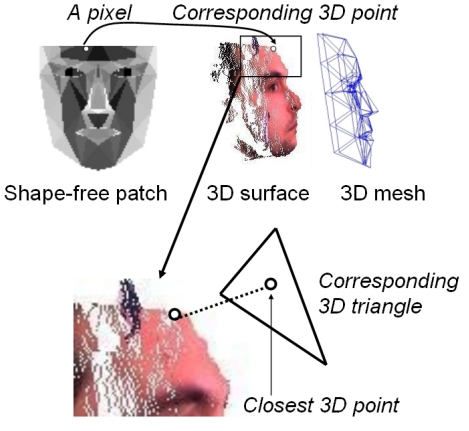


Fig. 5. At any iteration of the ICP algorithm, the computation of the closest points is carried out by exploiting the labelled 3D points and projecting them onto their triangles. This scheme can be very fast compared to the classical closest point computation.

pose. Many variants of the ICP algorithm have been proposed. A comparative study can be found in [10]. Figure 3 illustrates the main steps of the proposed approach. Notice that the stereo reconstruction only concern a subset of pixels that belong to the face region in the current image.

Since the monocular tracker provides the 3D face pose and the facial deformations by matching a warped version of the input texture with the facial texture model (both textures correspond to a 2D mesh), it follows that the out-of-plane motion parameters associated with the 3D face pose can be inaccurate even when most of the facial features project onto their true location in the image. We use this fact to argue that the use of the appearance-based tracker results and the 3D mesh model will greatly help the refinement step in the sense that i) it will provide the initial 3D pose required by the ICP algorithm, ii) it will

adapt the deformable model (mesh) to the current facial expression, and iii) it will make the search for the closest points simple and fast. Indeed, the most expensive step in any ICP algorithm is the computation of the closest points. In our case, we use a robust version of the ICP algorithm that registers the 3D mesh to the current facial surface. The provided registration will be considered as the 3D face pose improvement.

Figure 4 illustrates the basic idea that is behind the improvement step, namely the robust 3D registration. Figure 4.**a** illustrates an ideal case where the estimated appearance-based 3D face pose corresponds to the true 3D pose. In this case, the mapped 3D mesh is perfectly registered with the 3D facial surface provided by the stereo system. Here the mapping is provided by the appearance-based 3D face pose. Figure 4.**b** illustrates a real case where the estimated appearance-based 3D face pose does not correspond exactly to the true one. In this case, the improvement can be estimated by recovering the 3D rigid displacement $[\mathbf{R}|\mathbf{T}]$ between the 3D mesh and the current facial surface. To this end, we will use a robust ICP algorithm where the surfaces to be aligned are the current 3D facial surface and the 3D mesh - a piecewise planar surface. Note that both surfaces are expressed in the same coordinate system.

4.1 Computing the Closest Points

Any ICP algorithm should include the selection and matching of 3D points on the surfaces to be registered. In our case, the rough registration provided by the appearance-based registration will be used to get these correspondences rapidly. The process is illustrated in Figure 5. At any iteration of the ICP algorithm, the computation of the closest points is carried out using the following. A pixel in the shape-free patch is mapped onto the input range image in order to get the corresponding 3D coordinates on the 3D facial surface. Then, the obtained 3D point is orthogonally projected onto its corresponding 3D triangle in the 3D mesh (recall that the shape-free patch is a 2D representation of the 3D mesh). This scheme is very fast compared to the classical closest point computation. Once the set of 3D-to-3D points is computed we invoke a robust technique for estimating the 3D rigid transform between the 3D mesh and the facial surface. We point out that the use of robust techniques is required since some pairs will have large points-to-points distance due to possible self-occlusions associated with the 3D mesh. Robust ICP variants have been proposed in recent literature (e.g., see [13,14]). In our work, we use a RANSAC-like technique that computes an adaptive threshold for outlier detection. The tracking algorithm associated with one frame in the video is given in Figure 6. Notice that the number of points used by the robust ICP algorithm may vary from one iteration to another.

Computational cost. On a 3.2 GHz PC, a non-optimized C code of the whole approach takes about 1 second working on 1300 facial points with four iterations. This time can be considered as few hundred times faster than a classical ICP algorithm which usually takes between 5 and 10 minutes to register two facial surfaces.

1. Compute the 3D face pose and the facial actions using the appearance registration technique (see Section 4).
2. Iterate until the maximum number of iterations is reached or the registration error in two consecutive iterations is below a certain threshold: Compute a set of putative 3D-to-3D points between the 3D surface and the 3D mesh $\{\mathbf{S}_i \leftrightarrow \mathbf{M}_i\}, \quad i = 1, \ldots, N$ using the technique illustrated in Figure 5.
   Random sampling: Repeat the following three steps K times
   (a) Draw a random subsample of 3 different pairs of points. We have three pairs of 3D points $\{\mathbf{S}_i \leftrightarrow \mathbf{M}_i\}, \quad i = 1, 2, 3$.
   (b) For this subsample, indexed by k ($k = 1, \ldots, K$), compute the 3D rigid displacement $\mathbf{D}_k = [\mathbf{R}_k|\mathbf{T}_k]$, where $\mathbf{R}_k$ is a 3D rotation and $\mathbf{T}_k$ a 3D translation, that brings these three pairs into alignment. $\mathbf{R}_k$ and $\mathbf{T}_k$ are computed by minimizing the residual error $\sum_{i=1}^{3} |\mathbf{S}_i - \mathbf{R}_k \mathbf{M}_i - \mathbf{T}_k|^2$. This is carried out using the quaternion method [11].
   (c) For this solution $\mathbf{D}_k$, compute the median M_k of the squared residual errors with respect to the whole set of N points. Note that we have N residuals corresponding to all points $\{\mathbf{M}_j \leftrightarrow \mathbf{S}_j\}, \quad j = 1, \ldots, N$. The squared residual associated with an arbitrary point $\mathbf{M}_j$ is $|\mathbf{S}_j - \mathbf{R}_k \mathbf{M}_j - \mathbf{T}_k|^2$.
   Solution:
   (a) For each solution $\mathbf{D}_k = [\mathbf{R}_k|\mathbf{T}_k], k = 1, \ldots, K$, compute the number of inliers among the entire set of vertices (see [12]). Let n_k be this number.
   (b) Choose the solution that has the highest number of inliers. Let $\mathbf{D}_i$ be this solution where $i = \arg\max_k(n_k), k = 1, \ldots, K$.
   (c) Refine $\mathbf{D}_i$ using all its inlier pairs.
3. Based on the above 3D motion update the 3D face pose and go to 2.

Fig. 6. Tracking the 3D face pose and the facial actions in each video frame using appearance registration and a robust ICP algorithm

5 Experimental Results

Figure 7 displays the face and facial action tracking results associated with a 300-frame-long sequence (only four frames are shown). The tracking results were obtained using the adaptive appearance and the robust ICP algorithm described in Sections 3 and 4, respectively. The upper left corner of each image shows the current appearance model and the current shape-free texture. In this sequence, the nominal absolute depth of the head was about 80 cm. As can be seen, the tracking results indicate good alignment between the mesh model and the images.

Figure 8 displays the face and facial action tracking results associated with a 300-frame-long sequence (only four frames are shown). In this sequence, the nominal absolute depth of the head was about 65 cm. Figure 9 displays the 3D rigid motion computed by the robust ICP algorithm, that is, the expected improvement to the appearance-based 3D face pose (shown by a solid horizontal line). The improvements associated with the vertical translation and the pitch angle are quite small.

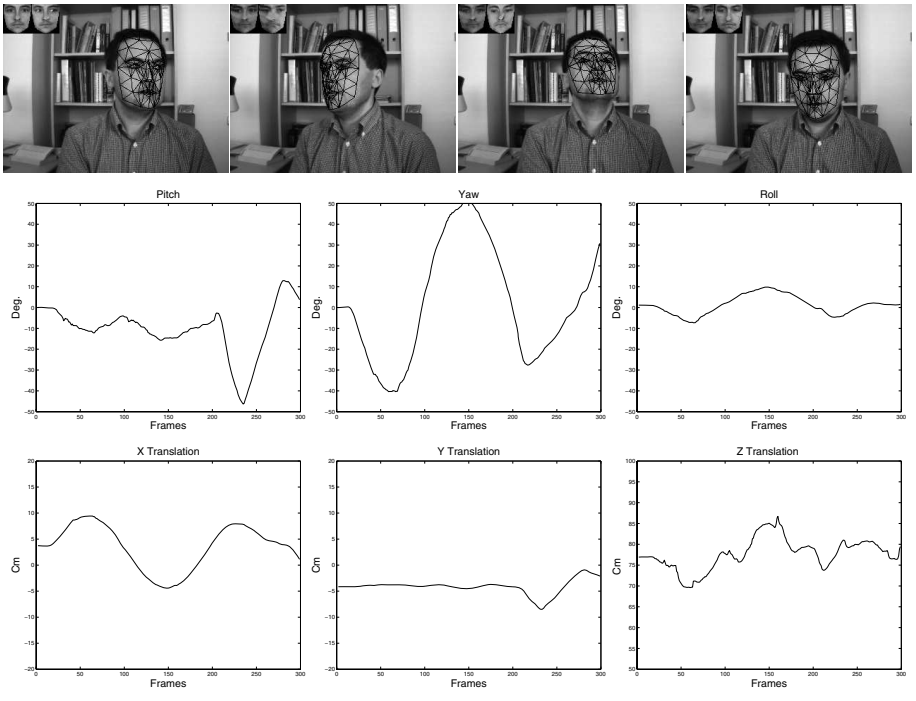


Fig. 7. Tracking the 3D face pose using the proposed technique. The sequence length is 300 frames. Only frames 38, 167, 247, and 283 are shown.

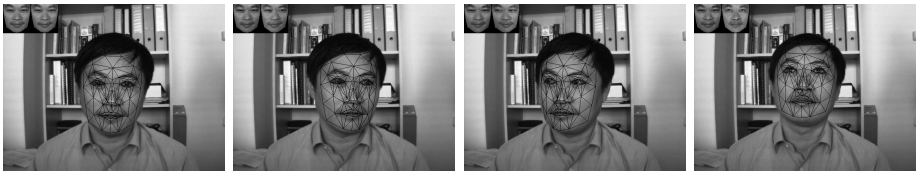


Fig. 8. Applying the developed tracker to a 300-frame long video sequence

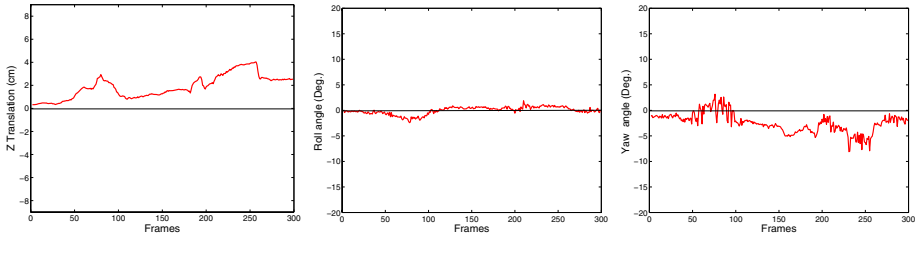


Fig. 9. The 3D rigid motions computed by the robust ICP algorithm associated with the sequence depicted in Figure 8: in-depth translation, roll angle, and yaw angle.

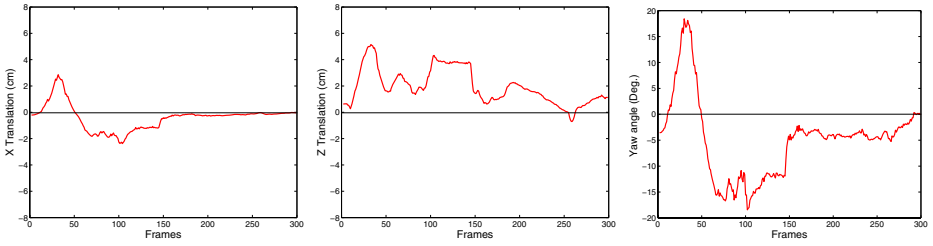


Fig. 10. Another sequence: horizontal translation, in-depth translation, and yaw angle.

Figure 10 displays the 3D rigid motion computed by the robust ICP algorithm associated with another 300-frame-long video sequence. In this sequence, the nominal absolute depth of the head was about 60 cm.

6 Discussion

In this paper, we have proposed a robust 3D face tracker that combines the advantages of both appearance-based trackers and 3D data-based trackers. We have shown that the use of an appearance-based registration together with a robust ICP has several advantages. The appearance registration has provided the facial action parameters as well as the initial estimate of the parameters for the robust ICP algorithm. On the other hand, the ICP algorithm has provided an accurate out-of-plane face motion. Due to the fact that the two surfaces are different not all degrees of freedom associated to the head pose can be improved. For applications that do not need high accurate out-of-plane head motions one can skip the use of the robust ICP algorithm. The proposed framework can be applied to tracking deformable surfaces as long as their deformations can be statistically modelled.

References

1. Moreno, F., Tarrida, A., Andrade-Cetto, J., Sanfeliu, A.: 3D real-time tracking fusing color histograms and stereovision. In: IEEE International Conference on Pattern Recognition. (2002)
2. Cascia, M., Sclaroff, S., Athitsos, V.: Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(4) (2000) 322–336
3. Ahlberg, J.: An active model for facial feature tracking. EURASIP Journal on Applied Signal Processing **2002**(6) (2002) 566–571
4. Matthews, I., Baker, S.: Active appearance models revisited. International Journal of Computer Vision **60**(2) (2004) 135–164
5. Malassiotis, S., Strintzis, M.G.: Robust real-time 3D head pose estimation from range data. Pattern Recognition **38**(8) (2005) 1153–1165

6. Besl, P., McKay, N.: A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(2) (1992) 239–256
7. Ahlberg, J.: Model-based coding: Extraction, coding, and evaluation of face model parameters. PhD thesis, No. 761, Linköping University, Sweden (2002)
8. Dornaika, F., Davoine, F.: On appearance based face and facial action tracking. IEEE Transactions on Circuits and Systems for Video Technology (To appear)
9. Dornaika, F., Sappa, A.: Appearance-based tracker: An evaluation study. In: IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. (2005)
10. Rusinkiewicz, S., Levoy, M.: Efficient variants of the ICP algorithm. In: IEEE International Conference on Pattern Recognition. (2001)
11. Horn, B.: Closed-form solution of absolute orientation using unit quaternions. J. Opt. Soc. Amer. A. **4**(4) (1987) 629–642
12. Rousseeuw, P., Leroy, A.: Robust Regression and Outlier Detection. John Wiley & Sons, New York (1987)
13. Chetverikov, D., Stepanov, D., Kresk, P.: Robust Euclidean alignment of 3D point sets: the trimmed iterative closet point algorithm. Image and Vision Computing **23** (2005) 299–309
14. Fitzgibbon, A.: Robust registration of 2D and 3D point sets. Image and Vision Computing **21** (2003) 1145–1153