Univ Access Inf Soc (2008) 6:363–374
DOI 10.1007/s10209-007-0096-6

LONG PAPER

Facial movement analysis in ASL

Christian Vogler · Siome Goldenstein

Published online: 3 November 2007

Abstract In the age of speech and voice recognition technologies, sign language recognition is an essential part of ensuring equal access for deaf people. To date, sign language recognition research has mostly ignored facial expressions that arise as part of a natural sign language discourse, even though they carry important grammatical and prosodic information. One reason is that tracking the motion and dynamics of expressions in human faces from video is a hard task, especially with the high number of occlusions from the signers' hands. This paper presents a 3D deformable model tracking system to address this problem, and applies it to sequences of native signers, taken from the National Center of Sign Language and Gesture Resources (NCSLGR), with a special emphasis on outlier rejection methods to handle occlusions. The experiments conducted in this paper validate the output of the face tracker against expert human annotations of the NCSLGR corpus, demonstrate the promise of the proposed face tracking framework for sign language data, and reveal that the tracking framework picks up properties that ideally complement human annotations for linguistic research.

C. Vogler (✉)
Gallaudet Research Institute, Gallaudet University,
800 Florida Ave. NE, Washington, DC 20002-3695, USA
e-mail: Christian.Vogler@gallaudet.edu

S. Goldenstein
Instituto de Computação, Universidade Estadual de Campinas,
Caixa Postal 6176, Campinas, SP 13084-971, Brazil
e-mail: siome@ic.unicamp.br

1 Introduction

With the growing popularity of speech and voice recognition technologies, it is only a matter of time before they pose serious accessibility problems to deaf people who use signed languages as their primary mode of communication. Sign language recognition technologies are, therefore, an essential component of accessible human–computer interaction (HCI). Research into recognizing the manual components of sign languages has yielded promising results [1, 35], but to date, this research has virtually ignored the facial expressions that arise as part of a natural sign language discourse, even though they carry important grammatical and prosodic information [24]. In addition, linguistic research into signed languages can greatly benefit from face tracking and recognition systems, as they can assist with the tedious task of annotating data.

One reason why facial expressions have been ignored to date is that the challenges in incorporating them into a recognition system are substantial. Tracking human faces from video is a very difficult problem, even more so because 3D information integral to signed languages needs to be recovered, such as head tilting and side-to-side movements. In addition, the subtle facial movements that arise in signed languages require a much greater degree of precision than many past tracking approaches have been able to provide. Another complicating factor is that the system must be able to track and recognize the facial expressions exactly as they would occur in natural settings. This requirement precludes many controlled laboratory conditions that would otherwise simplify the task, because they would alter the appearance of the signs. Foremost among them is a lack of control over the movements that the subjects exercise. As a result, it is necessary to deal with extreme head movements from side to side, frequent

self-occlusions of the face by the signer's hands, as well as possible partial obstructions of the signer's facial features by hair.

This paper presents a 3D deformable model tracking system and applies it to American Sign Language (ASL) sequences of native signers, taken from the National Center of Sign Language and Gesture Resources (NCSLGR) [25]. Deformable models restrict the family of possible solutions. Instead of estimating the positions of every relevant feature of the face, the approach constrains the changes in the image to changes in the values of a 3D parametrized model that describes, in the case of face tracking, the subject's face. These parameters model both the rigid motion (orientation and translation), as well as nonrigid deformations, such as eyebrow, lip and jaw movement (Sect. 3).

Deformable model tracking is an inductive procedure (Sect. 3.1). The basic underlying assumption is that the parameters that register the 3D model to the image in the previous frame are known. Then, based on changes in the new frame, the algorithm locally searches for the new configuration of the parameters that align the model to the new image. Since a human face has locally distinct photometric properties, deformable model applications use various techniques to find features that define local correspondences between the images and the model: point trackers for image corners, edge trackers for image borders, and optical flow for areas that have texture. A statistical framework then merges these correspondences (Sect. 3.3).

In their basic formulation, deformable models are sensitive to outliers; that is, some of the image features may provide wildly inaccurate estimates and throw-off the tracking process as a whole. Eliminating the outliers is essential, especially in the case of sign language sequences, because self-occlusions of the face by the hands generate large numbers of them (Sect. 3.4.2).

Originally, these methods were all geared toward tracking the face from noisy images and had nothing to do with sign language. This paper shows through experiments on the NCSLGR data (Sect. 4) that the same methods can be used to deal with uncertain information on the subject's face in sign language recognition applications, especially occlusions, and that 3D deformable model tracking holds great promise for both sign language linguistic research and recognition in the context of HCI. In particular, the experiments validate the output of the face tracker (Sect. 4.1) against expert human annotations of the NCSLGR corpus (Sect. 4.2). In addition, plots of the extracted face trajectories exhibit several interesting properties that are not picked up in the annotations, showing that linguistic research and computer science research into signed languages complement each other (Sect. 4.3).

2 Related work

There are many approaches to deformable model tracking, such as snakes [21], active shapes [8], active appearance models [7], and active contours [2]. Other methods include a local-global hybrid approach [22], and PCA (principal component analysis) decomposition [3]. PCA decompositions are an extremely powerful way to fit, track and even recognize objects [3, 10, 23, 26, 27,], at the cost of requiring a large database of examples. Additionally, a powerful deformable volumetric model has been used for fast face tracking [33] and subtle motion capture [37]. Predictive filters [16], such as the Kalman filter, can add reliability to these tracking methods [18].

In a broader sense, current approaches can be divided into two categories: the ones that use machine learning techniques to train a model on a set of data before they can be put to work, and the ones that do not. The former category often requires carefully constructed training examples, which are not always available in preexisting data sets. The latter category is more flexible in this respect; however, it still requires that models are fitted to their respective starting frames, which can be a time-consuming task. A possible step toward automation of the fitting task consists of combining stereo with shading [29], and using anthropometric databases to generate biometrically accurate models [9].

This paper's deformable model tracking framework requires no training. As a result, it is particularly suitable for the analysis of sign language data that were not collected with training in mind. Parametrizing muscle actuator groups [13] provides another way to recognize dynamic facial expressions without training. A contrasting approach to analyzing facial expressions as part of language recognition is based on training active appearance models [6].

Recognizing facial expressions is just one part of a comprehensive sign language recognition framework. Recent work has focused on modeling the language in terms of its constituent parts (phonemes) [1, 35] and first steps toward signer-independent recognition [14, 36, 39]. Other work includes 3D reconstruction of the hands [11], and recovery of facial action units [38].

3 Deformable models

For sign language recognition and analysis, it is necessary to track a moving face. Deformable models provide a way to reduce the dimensionality of the tracking problem. Working with a free-form 3D mesh would require obtaining the position of every single node on the mesh in some way. In contrast, within a deformable model, the position of every node on the mesh is specified through a function

of a small set of parameters—the parameter vector **q**. The components of this vector can represent such things as the position and orientation of the model in 3D space, the degree of eyebrow raising, mouth aperture, and so on.

3.1 Mathematics of deformable models

Mathematically, there are many ways to represent deformable models. A simple, yet powerful, approach consists of using a triangulated 3D mesh, with a function for each of its nodes. This function evaluates each node's position for any given value of the parameters, such as the eyebrows, face orientation, and the jaw opening. Sects. 3.2.1 and 3.2.2 describe a directed acyclic graph representation [17] and an adaptive resolution method [19] to construct this mesh.

Tracking is an inductive process. For the initial frame, the parameters that put the face in its proper place in the image, with the correct facial expressions, must be known—via either manual or automated methods of fitting the model to the image. For every new frame, the tracking system needs to adjust the parameters, such that they best describe the model for this frame. The basic idea is to select distinctive image features, such as edges and corners, and to track them over time using standard 2D image processing methods. The result of this process consists of the (potentially inaccurate) positions of where the model nodes should be in the new frame. The system then finds the 2D displacements between the current model node positions—through a projection of the model onto the image—and the tracked features, and express them as vectors, called $\mathbf{f}_i$. Then, it performs a local nonlinear optimization on the parameter vector **q**

$$\dot{\mathbf{q}} = \sum_i B_i^{\top} \mathbf{f}_i, \tag{1}$$

where B_i is the Jacobian

$$B_i = \frac{dp_i}{d\mathbf{q}} = \begin{bmatrix} | & | & & | \\ \frac{\partial p_i}{\partial q_1} & \frac{\partial p_i}{\partial q_2} & \cdots & \frac{\partial p_i}{\partial q_n} \\ | & | & & | \end{bmatrix}, \tag{2}$$

to find the value of **q** that minimizes the 2D displacements. The simplest way to accomplish this task is through an iterative gradient descent method, using Eq. (1) to find the appropriate direction of the next step [17].

3.2 Representation of deformable models

In computer science, there is generally a big gap between the mathematical description of something and its actual implementation. Deformable models are no exception, and the design decisions made during the application development affect the families of deformations and models it can express. The deformable model representation problem is far from solved, and this research straddles the boundaries across computer vision, computer graphics, and numerical analysis. Good representations should be able to describe any geometry and its deformations, while still allowing fast computations of positions and Jacobians, with a small memory footprint.

3.2.1 Directed acyclic graph representation

The most basic representation of a deformable model describes every point on its surface explicitly. Each such point is computed through a different function on the parameter vector **q**. This function is responsible for the calculation of the point's position, and Jacobian, for every possible parameter configuration. With m points and n controlling parameters ($q_1 \ldots q_n$, that can be stacked in a vector **q**), the model is composed of the totality of the functions

$$\begin{cases} p_1 = F_1(q_1, \cdots, q_n) \\ \qquad \vdots \\ p_m = F_m(q_1, \cdots, q_n) \end{cases}, \tag{3}$$

m functions on $\mathbb{R}^n \rightarrow \mathbb{R}^3$ altogether, and their respective Jacobian matrices.

One of the most flexible and powerful ways to store these functions is via directed acyclic graphs (DAGs), which describes the structure of each of these functions [17]. The DAG contains nodes of three different types: parameters, positions, and mathematical operations. The directed edges represent the flow of information and the dependency chain. A dependency arises when the position of one model point depends on another one; for example, the points along the contour of the lips depend on the positions of the mouth corners.

In the DAG, parameter nodes each represent a scalar parameter that controls a specific aspect of the model. The parameter vector **q** consists of the collection of all such scalar parameter values. As they constitute the fundamental entities that control the shape of the model and are adjusted to fit the images during the tracking process, they do not depend on any other nodes.

Position nodes represent 3D positions, which can be either model points or intermediate values in more complex mathematical expressions. Each position node depends on exactly one mathematical operation node, which defines its generating function.

Mathematical operation nodes are the actual calculating engines, and can depend on any number of position and

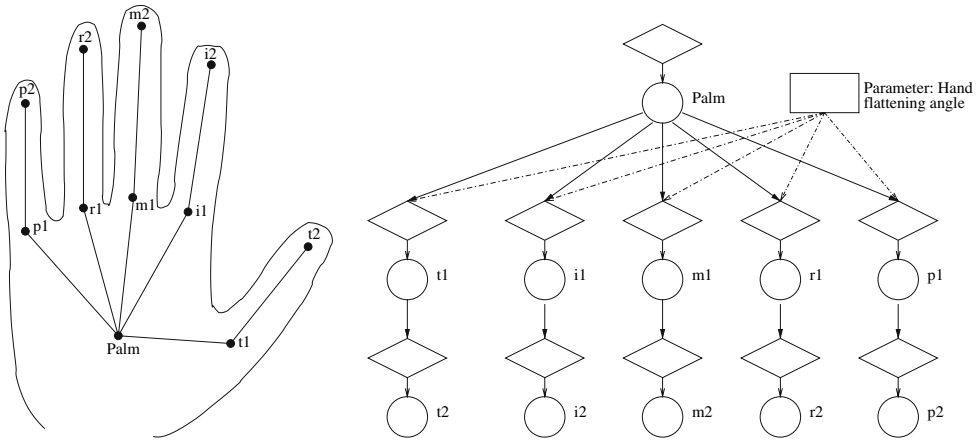

Fig. 1 A DAG for a simple eleven-point stick representation of an hand. In this representation, there is one parameter node that controls by how much the fingers are flattened at the metacarpophalangeal joints. Hence, the positions of the finger nodes depend on both the position of the palm and the hand flattening angle; these contributions are combined in the mathematical operation nodes that are denoted by the *upper row of diamonds*

parameter nodes. They perform the calculation of a model point location and its Jacobian. Figure 1 illustrates a rather simple DAG, with one parameter node, that represents a simple eleven-point articulated hand, useful in tracking applications [4]. In this figure, circles denote position nodes, squares denote parameter nodes, and diamonds denote mathematical operation nodes.

3.2.2 Deformation map representation

The representation of a deformable model via a DAG is fast and general, but it is also inflexible. In many computer graphics applications, it is necessary to refine the model, adding more detail and creating new points on the model dynamically. The DAG does not provide a natural way to describe the function of these newly created nodes.

A better representation allows the dynamical creation of model points, as well as their respective deformation functions, by defining layers of deformations on a $\langle u, v \rangle$ 2D parametric space. This $\langle u, v \rangle$ space spans the surface of the 3D model, with u, v describing where on the surface a point is located. This representation can define a deformation function that depends on the value of the parameters $\mathbf{q}$ and u, v, so

$$P_i = F(\mathbf{q}, u, v) \tag{4}$$

instead of $P_i = F_i(\mathbf{q})$, as in Eq. (3). Figure 2 shows how a few parameters that affect local regions of a base mesh can generate deformations in a 3D face model.

3.2.3 Statistical extensions to the framework

One problem with using features from different image processing methods (such as corners, edges, optical flow) is that there is no straightforward way to integrate information from different sources ("cues"). However, earlier work proved that the distribution of $\dot{\mathbf{q}}$ can be properly approximated as a multivariate Gaussian [17], by assigning regions of confidence to the 2D displacements $\mathbf{f}_i$, and representing them as affine forms[32], which are symmetric convex regions. This statistical representation of $\dot{\mathbf{q}}$ results in a maximum likelihood approach to the fusion of the information from different cues [17], and to a confidence measure of the distribution of the final model parameters, represented as a Gaussian probability distribution. Knowing this distribution is important for two reasons. First, it provides the groundwork for using a Kalman filter to stabilize the tracking [18], and second, it

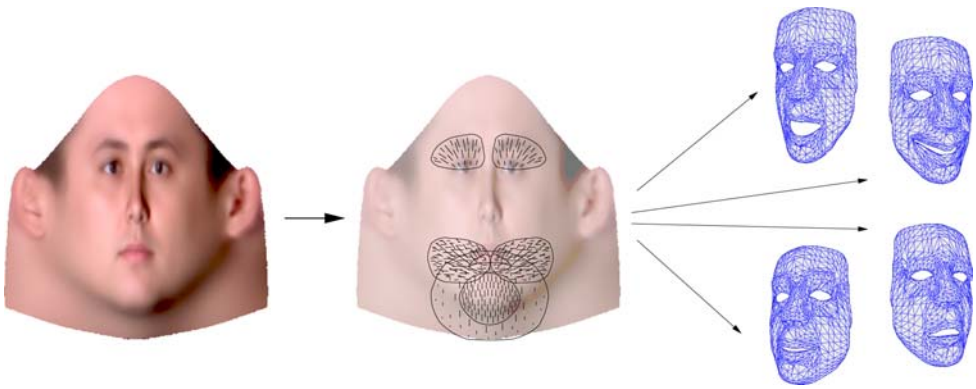

Fig. 2 Deformation maps: with a set of parameters, and localized areas of effect, we can create a 3D deformable face model independent of the resolution of the mesh

provides the necessary information for identifying outliers among the features. Both these aspects are important for dealing with occlusions.

3.2.4 Outlier rejection and occlusion handling

Deformable model tracking depends on the quality of low-level correspondences, and uses well-known computer vision algorithms to find and track corners [30], calculate the optical flow [31], and detect edges [5] to establish the 2D displacement forces for the optimization procedure described in Eqs. (1) and (2).

Unfortunately, vision low-level correspondence algorithms sometimes return strange results. Even in the best-case scenario, the results from the low-level correspondences are corrupted by noise. Statistical modeling of the errors caused by noise can handle these corruptions, and aggregating a large number of contributions cancels out the effect of noise. Even so, frequently, completely incorrect correspondences occur that would throw-off the results of the optimization stage by an arbitrarily large amount, causing the 3D model to go out of alignment with the video. Such outliers can happen, among other reasons, because of incorrect assumptions by the low-level algorithms about the characteristics of the images, or specifically because of occlusions. If the hand occludes the face, often a tracked feature "jumps" from the face to the hand, which subsequently pulls the feature away from the correct location in the face. Figure 6 shows a characteristic example of this phenomenon.

3.2.5 Outlier rejection in deformable model tracking

Since there is no closed-form algebraic solution for the deformable model tracking optimization problem in Eq. (1), it is not possible to apply some of the traditional outlier rejection techniques directly, such as RANSAC [15]. Instead, all 2D displacements are projected into the parameter space, and a Gaussian approximation to their distribution is found. Applying a robust estimator, called MCD (minimum covariance determinant) [28], solves the problem of outlier masking, common when using the standard Gaussian maximum likelihood estimators for the mean and covariance matrix. Conceptually, MCD finds the subset of features that minimize the spread (i.e., the covariance of the Gaussian model). The proposed method then tests the overlap of each individual feature, each of which also has a statistical description, with the Gaussian model, and rejects each feature whose statistical distribution is incompatible with this model. The details of this method can be found in Refs. [20, 34].

The breakdown point of MCD depends on the number of features that are chosen as representative of the Gaussian model. This parameter is configurable by the application, and can be tailored to the needs of sign language tracking, depending on the expected number of outliers.

3.2.6 Occlusions as outliers and fast hand movements

As mentioned before, in sign language, the hand frequently occludes the face (Fig. 4). This scenario has to be handled gracefully without ever causing catastrophic results, such as a loss of track. From a high-level point of view, an occlusion means that no information on the hidden regions of the face is available, and hence any image features that are affected by these regions must be treated as suspect. Identifying these regions themselves is not easy. However, the suspect feature points typically express behavior that is inconsistent with the overall flow of the model; that is, they become outliers.

However, modeling occlusions as outliers makes the assumption that the number of good correspondences outweighs the number of bad correspondences in the context of the Gaussian movement model; that is, there are less than 50% outliers. In the case of fast hand movements in front of the face this assumption is not valid, because large swathes of the face are occluded in successive frames, and the correspondences conform to two separate movement models: the facial movement and the hand movement. In fact, for this reason, the proper approach would be to use a multimodal representation for the movement that separates the hands from the face, but making such a separation work is difficult in practice.

As this paper only deals with facial movement, it is not necessary to go so far as to attempt a separation. All that is needed is an initial rough filtering out of the correspondences that clearly belong to the hand movements before applying the outlier detection mechanism, so as to push the ratio of good-to-bad correspondences over 50%. To this end, the proposed method uses a Bayesian optical flow algorithm [31] to remove regions whose movement is considerably faster than the general face movement.

3.2.7 Side effects of occlusion handling

Occlusions introduce an extra degree of complexity in deformable model tracking. When deformations act only on local model regions, an occlusion might hide the entire area of such deformations, and make it impossible to find any valid observation. For instance, it is impossible to infer much about the mouth if nearly its entire area is hidden by the hands.

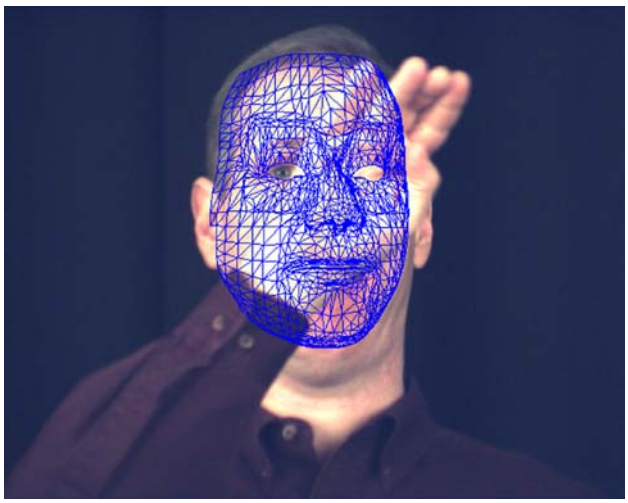

Fig. 3 Occlusions hide parts of the model, and make the estimation of some parameters an almost impossible task

This problem of not being able to observe the effect on a parameter during an occlusion leads to severe numerical issues that are far from being solved. The deformable model tracking Eq. (1) and (2) assume that all parameters are given equal consideration by the 2D correspondences, but if many of them are screened out due to occlusions, this assumption no longer holds. The effect of this scenario is that sometimes local deformations, such as eyebrow movements or mouth movements are not properly picked up by the 3D model during tracking. Figure 3 illustrates a quite difficult situation during face tracking, where very few areas of the face are visible.

Normalizing the contribution to each model parameter by the number of available 2D correspondences that affect each respective parameter [20] is an good first step toward a solution. However, taking this measure still does not solve the problem when there are so few available correspondences that the estimation of a model region becomes underconstrained and there are multiple solutions with different parameter vector that all fit the given observations of the correspondences in the image. Thus, mitigating the effect that occlusions have on parameter estimation in localized face regions is still an open research area.

Finally, detecting and eliminating of outliers is not enough by itself. It is also important to reacquire facial features after the occlusion is over, which in itself is far from being an easy problem.

4 Experiments and discussion

Past work has validated the efficacy of statistical cue integration and outlier rejection within the tracking framework extensively [17, 34]. In addition, the suitability of the framework for recognizing facial expressions was demonstrated in stress detection experiments [12], where the computer achieved a detection accuracy of 75–88% across 60 subjects in double-blind experiments.

Because of the high speed of the hand and facial movements, as well as the frequent self occlusions discussed earlier, tracking sign language sequences is substantially harder than the previous stress detection task, and even harder than tracking corrupted videos. To make matters worse, validation of sign language sequences is a hard task itself, because it is not possible to use any kind of intrusive system to measure the subject's head and facial movements. Doing so would interfere with the subject's production of signs and substantially alter the appearance of the utterances, thus violating the prerequisite of having natural utterances from native signers.

As a result, the best that can be done is to compare the output of the tracking process with the annotations provided by expert human transcribers. To this end, this paper experiments tracked sequences taken from the NCSLGR corpus. This corpus consists of a large number of videos of native signers taken in a carefully constructed laboratory setting to ensure that these sequences were representative of ASL, along with annotations.

The video in Fig. 4 is representative of the types of facial images and stories that can be found in the corpus. It shows a story about a close call with a deer on a highway. This sequence, just like the others, was recorded in full color at a resolution of 640 × 480 at 60 frames per second. The annotations consisted of information on head rotation and tilts, along with eye blink information and glosses.[1]

4.1 Tracking results

To test the hypothesis that the proposed methods to deal with noisy images also help with sign language tracking, especially self-occlusions, video clips were tracked on three different subjects. For each subject, the shape of the generic 3D face model needed to be fitted to the characteristics of that person's face in a semi-automated step. This step consisted of choosing a frame with a frontal view, and selecting correspondences between the nodes on the model and the image. The system was then integrated according to Eq. (1) from Sect. 3.1, with the shape characteristics functioning as the model parameters, and the model-image correspondences acting as the $\mathbf{f}_i$ in this equation. The integration process subsequently yielded a version of the model that was adjusted to the particular subject's face. Although this is a laborious task, it needs to

[1] Glosses are representations of the signs by their closest English equivalent in all capital letters.

Fig. 4 Excerpt from the tracked sign language sequence ("Close Call," taken from NCSLGR), showing the sign for "REMEMBER." Occlusions of the face, as shown in the middle image, are frequent and greatly complicate the tracking problem

be done only once per subject. Afterward, the model can be applied to the same subject shown in arbitrary poses on arbitrary video clips.

The face models consisted of 1,101 nodes and 2,000 triangles. Tracking the video sequences with these models, using gradient descent with 600 iterations per frame, took 0.3 s per frame on an AMD Opteron 246 workstation running a 64-bit version of Linux. This number includes time spent on image processing, which still holds much potential for performance optimizations.

Figure 5 shows some representative tracking results. Note that particularly in the bottommost two examples, there are strong self-occlusions of the face, yet they have no effect on the alignment of the face model. The explanation for why the model is unaffected by the occlusions lies in the behavior of the outlier rejector, which is described briefly in Sect. 3.4.2. The following discussion focuses on the effect of outlier rejection on point features, but a similar argument also applies to other cues, such as edges.

Figure 6 shows three snapshots of the point tracker, with the tracked features divided into acceptable points and outliers, at the frame immediately before the occlusion, during the occlusion, and immediately afterward. Many points are already filtered out by the Bayesian optical flow preprocessing step (Sect. 3.4.2), and others are dropped by the point tracking algorithm, because it cannot find any acceptable matches for them during the occlusion. Even so, some points "survive" and are pulled along the contour of the hand (Fig. 6, center). These points then stay on the hand contour and are pulled downward (Fig. 6, right). They are now gross outliers, and if they were not recognized as such, they would pull the face model downward, thereby destroying the alignment between the model and the video.

Intuitively, the effect of robust parameter space outlier rejection is to select a relatively small subset of features that forms the most likely hypothesis for the trajectory of the model parameters, and to discard any point that does not support this hypothesis. Under these circumstances, the points on the hand contour in the center and right parts of the figure are clear outliers, because they induce a downward motion that is incompatible with the general sideways and tilting motion of the subject's face, as well as other points on the face that have provided unreliable information.

Dealing with partial occlusions in this manner works well, as long as the percentage of gross outliers does not rise above the breakdown point of the MCD estimator. Earlier work showed that the optimal trade-off between robustness and efficiency mandates a breakdown point of 75–80% [34], which roughly corresponds to a maximum allowable outlier percentage of 25%.[2]

4.2 Data analysis and comparison with corpus annotations

The sequence of parameter vectors $\mathbf{q}$, obtained through the tracking process, could conceivably be used as features for a recognition algorithm. For a comparison with the expert human annotations, the head rotation information contained in this vector first needs to be converted to Euler angles. In Euler nomenclature, the side-to-side head movement corresponds to the heading (yaw), the forward–backward tilt corresponds to the pitch, and the side-to-side tilt corresponds to the roll (cf. the mention of the NCSLGR corpus and annotations above). Representative plots of these three head angles, in comparison to the annotations, are shown in Figs. 7 and 8.

These plots exhibit qualitative agreement between the tracked head parameters and the human annotations. There are slight discrepancies in the boundaries, but these are difficult for humans to catch accurately. Future work will have to do further validation to determine whether the discrepancies are due to tracking errors or human error, and whether they will negatively affect automated recognition of facial actions.

In general, the human annotations have quantitative counterparts in the plots, such as positive versus negative angles, and magnitude of angles, which holds great promise for future recognition work. Anticipatory head movements, characterized by "start" and "end," are an exception, because they are impossible to quantify from the angles. In fact, with respect to head movements these two

[2] The exact number is dependent on the dimension of the parameter space; higher dimensions reduce the percentage.

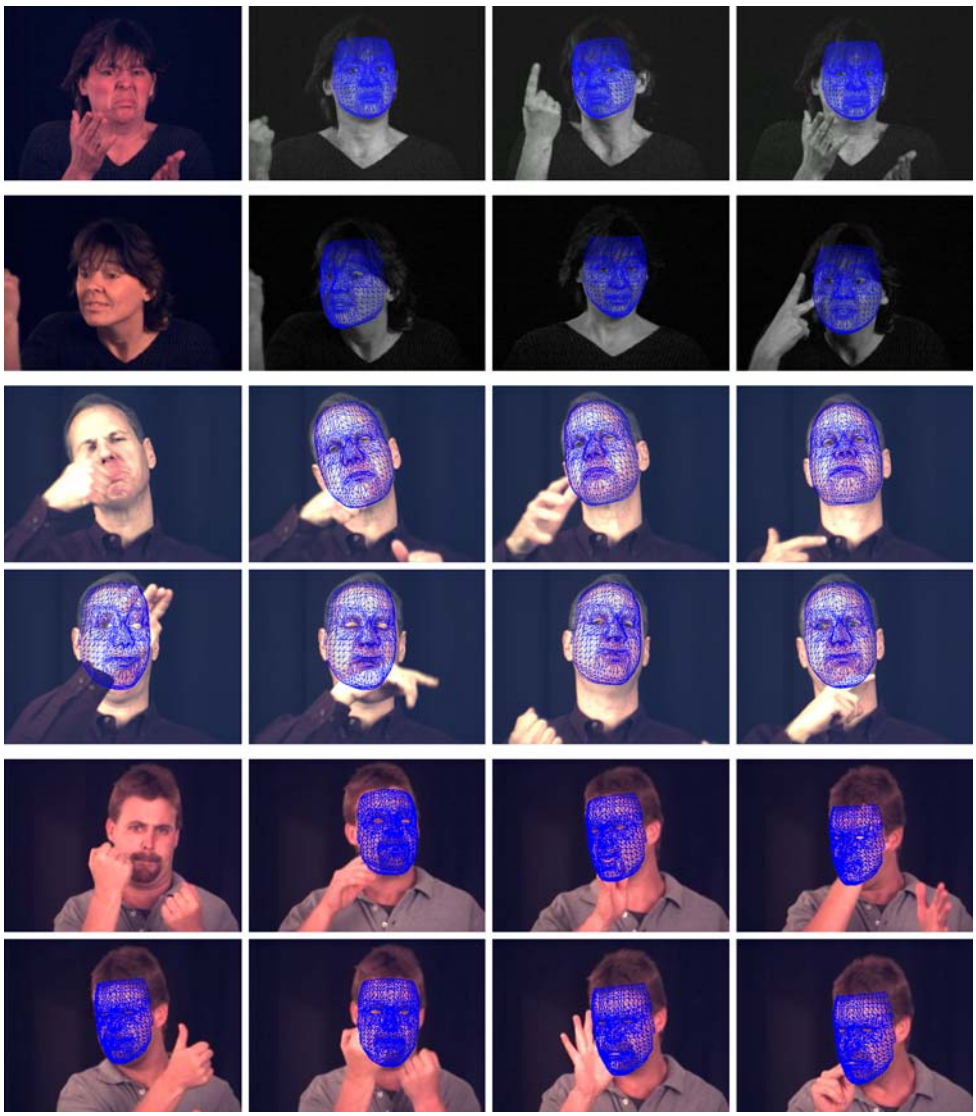

Fig. 5 Four tracking examples from three different native signers with the tracked 3D model overlaid. The self-occlusions of the face by the hands have no effect on the tracking

labels are highly subjective for humans (although much less so for other types of movements).[3] Essentially, they mark the head moving into position, immediately before and after a movement starts being perceived as significant by the human annotators.

Also of note is that there is remarkably little jitter in the plots, considering that the extracted angles were not smoothed in any way. A large part of this absence of jitter is due to the outlier rejection method, which, as previously stated, eliminates features that do not coincide with the overall tracking hypothesis.

The plots, however, exhibit clear peaks and valleys in areas that are perceived as a single entity by the human annotators, such as the two valleys in the head pitch plot in Fig. 7. These highlight how human annotators make decisions about what to annotate in particular, and what they concentrate on.

4.3 Implications for linguistic and recognition research

The differences between the human annotations and the plots lead to several interesting consequences and problems for both linguistic and recognition research. On the linguistics side, fundamentally, the human annotators have to deal with a small discrete set of labels to describe continuous phenomena, such as the degree of head tilt, which drops information on the exact kinematics.

As a consequence, the annotations often focus on the markings that the annotators perceive as important for their particular research project, whereas in reality there are complex interactions with the movements accompanying lexical items (i.e., individual signs).

[3] Carol Neidle, personal communication.

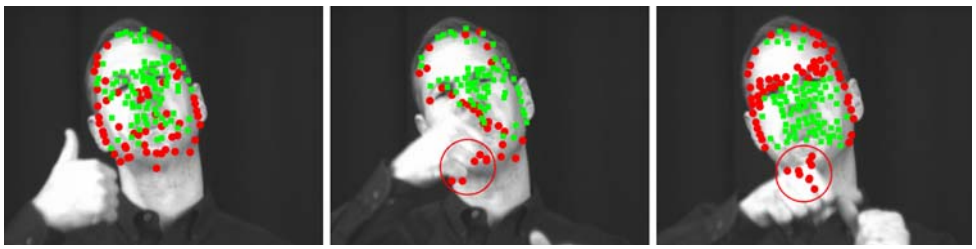

Fig. 6 Outlier rejection behavior during occlusion. Outliers are plotted as *red circles*, whereas valid points are plotted as *green boxes*. *Left*: Immediately before occlusion. *Center*: During occlusion. *Right*: Immediately after occlusion. Some points are erroneously "acquired" by the hand, but rejected as outliers (*large circle center and right*)

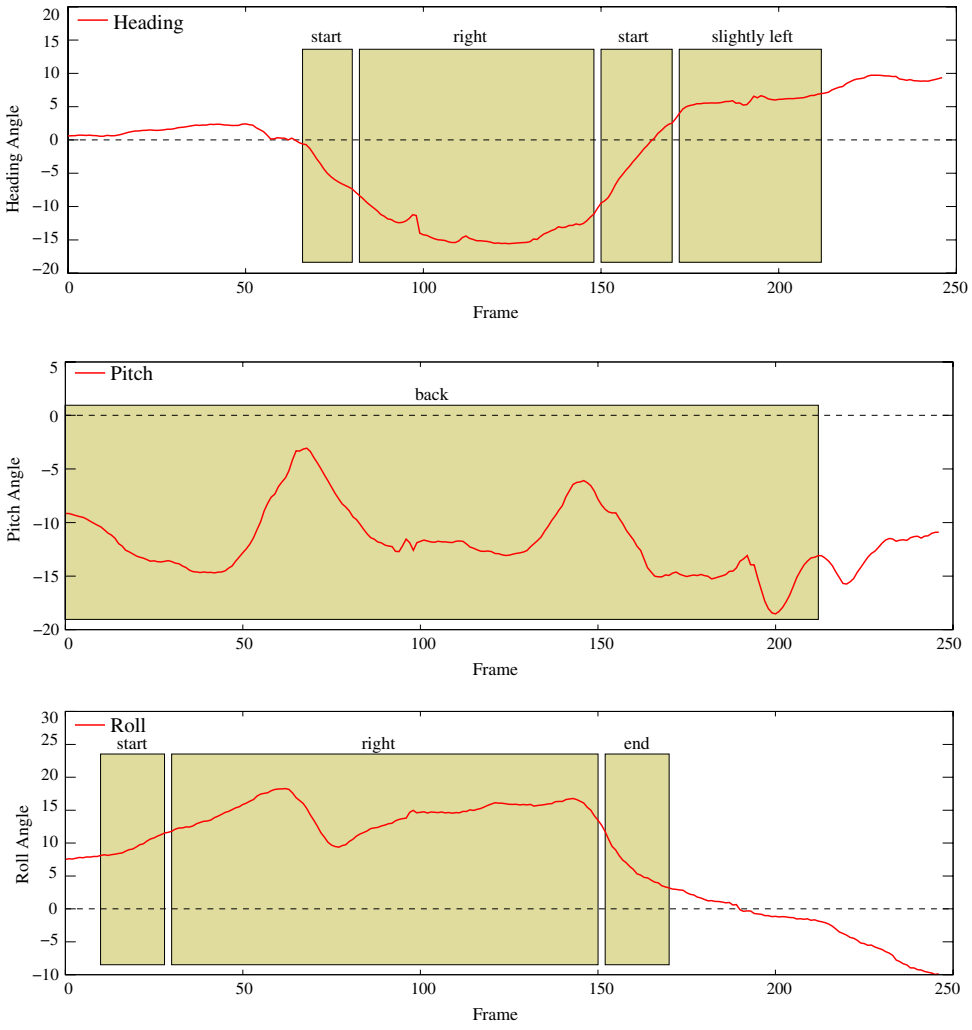

Fig. 7 Plot of the head's Euler angles compared to the expert human annotations of the NCSLGR (*labeled boxes*) for the first sentence of the "Close Call" sequence. "Start" and "end" correspond to anticipatory head movements. In the top figure, "right" and "slightly left" indicate the respective head turns; in the middle figure, "back" indicates a head tilt backward; and in the bottom figure, "right" indicates a head tilt to the right

Although humans could attempt to pick up the finer details of the movements, and to transcribe them as well, doing so would require an inordinate amount of time and effort, especially because humans have a hard time judging 3D angles on 2D video. Moreover, such an effort would not solve the basic problem that the annotators have only a discrete set of labels at their disposal.

Thus, for a linguistics researcher wishing to investigate the interplay of overlapping head movements, the tracking framework can provide an invaluable research tool. The main hurdle to providing this tool lies in devising a suitable user interface. At present, effective 3D tracking requires considerable computer science expertise, and it is not clear how to overcome this obstacle in the short term. For the

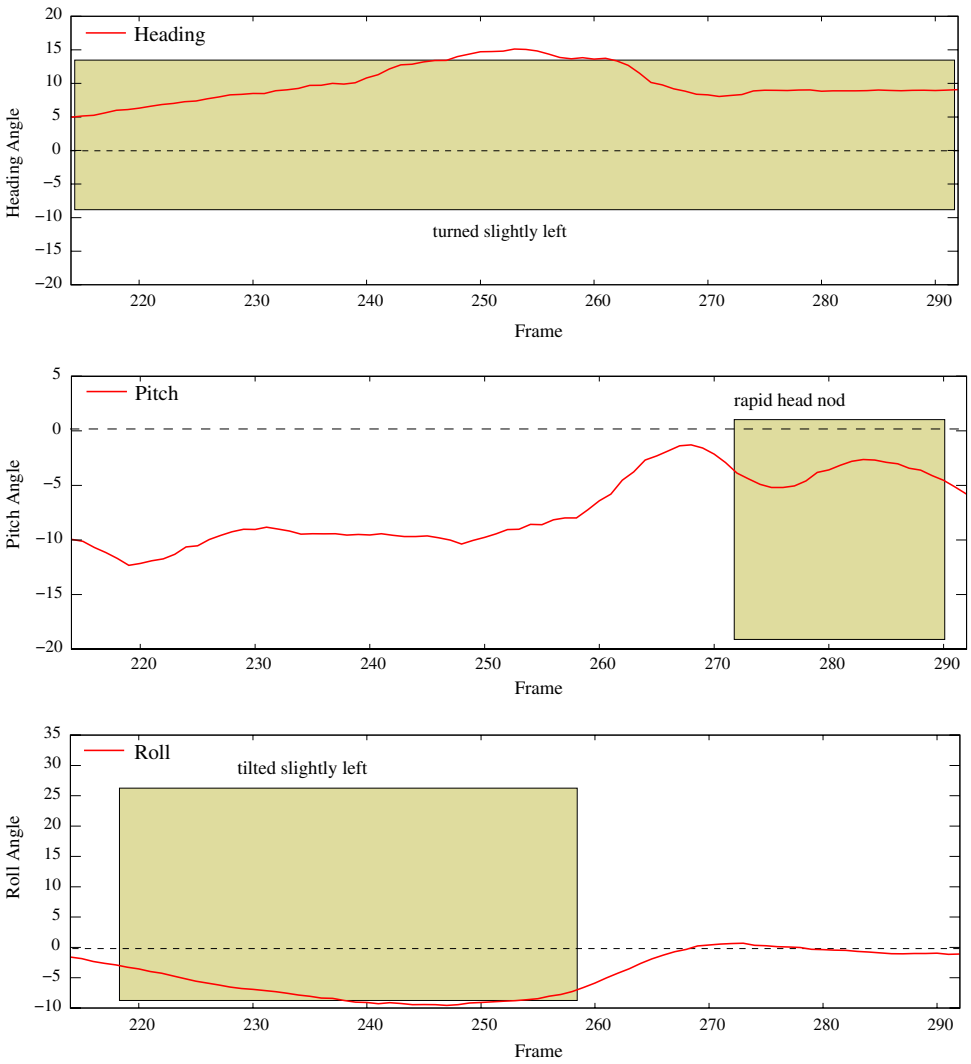

Fig. 8 Plot of the head's Euler angles compared to the expert human annotations of the NCSLGR (*labeled boxes*) for the second sentence of the "Close Call" sequence. In the middle figure, the rapid head nod" corresponds to the dip in the pitch angle. In the top and bottom figures, the tracked angles correspond to the human annotations, as well. However, in all three figures there is some disagreement as to where exactly the events start and end

time being, collaboration between sign language linguists and computer scientists remains the most promising avenue of action.

On the computer science side, the clear correspondence between the head angles and the head rotation and tilt labels holds great promise for future systems that recognize the nonmanual markings of signed languages. It is important not to get hung up on the question of identifying the anticipatory head movements (i.e., the portions labeled with "start" and "end"), which only exist to provide the human annotators with a greater choice of discrete labels. Rather, the first step for a recognition system should be the segmentation of the head movements into meaningful parts.

The fluctuations in the angles greatly complicate developing an automated recognition system. For instance, given the task of extracting the segment for the head tilt backward in Fig. 7, in accordance with the human annotations, if a recognizer naively thresholded the angles to label the degree of tilt, it would incorrectly interpret the sequence as alternating between slight and full tilts backward, whereas conceptually, it consists of a single, long full tilt. In a similar vein, the rapid head nod in the pitch plot in Fig. 8 is obvious to humans, because it shows up as a short dip between otherwise level sections of the head pitch angle. Yet, how would an automated recognizer distinguish this situation from the one in Fig. 7, where there is a dip of a longer duration that is of no semantic consequence?

What humans perceive conceptually does not always agree clearly with the actual kinematics of the head movements. At this stage, there are several explanations. First, the human annotations could be inaccurate. Estimating and quantizing 3D movements from 2D video is difficult, especially when the human annotators pay more attention to the discrete aspects of the head movements than the continuous aspects.

Second, the tracking process could be inaccurate. In order to confirm or eliminate this possibility, further

extensive validation of the sign language sequence tracking is needed, but doing so is a very difficult task, because the ground truth comes from human annotations, which themselves are prone to inaccuracies and errors.

Third, as numerous experiments on human perception have shown, humans tend to infer conceptual things into images that are not physically there. It is possible that the annotators' understanding of the ASL stories has biased them toward inferring certain head positions or movements that are typically associated with the respective signs in the stories, even if they do not appear clearly on the video. In addition, the annotations in question are preliminary and have not yet been validated.

Whatever is the cause of the discrepancies between the human annotations and the kinematics measurements from the face tracker, a recognizer will need to look at the global behavior of the head, not only localized angles, to distinguish between inconsequential movements and movements that carry semantic meaning.

5 Conclusions and outlook

The methods that make face tracking robust against noisy images carry over well to meet the demands of face tracking in the context of sign language analysis and recognition. Outlier rejection, in particular, makes the tracking framework resistant to partial face occlusions, as long as enough valid image features remain to keep the percentage of outliers below the breakdown point of robust statistical estimators. Outlier rejection alone fails in the presence of total occlusions, but preprocessing images to drop features that coincide with fast hand movements, through calculating the optical flow field, works surprisingly well even with near total occlusions.

In the end, the goal should not be accurate tracking during full occlusions, but rather graceful recovery from such events. For this recovery to work, a reliable way to reacquire facial features that were lost during an occlusion is required. At present, such a reacquisition is still an open problem.

This tracking framework can be used as a basis for assisting sign language linguistic research. The ability to extract and plot the trajectories of various facial parameters may well prove invaluable for research into sign language prosody. In addition, it could help with the verification of annotations. Moreover, tracking the human face is an important first step toward augmenting a sign language recognition system with facial expressions. The next step is tracking a large number of sequences, so that the extracted parameters can be used in machine learning algorithms.

Acknowledgments The research in this paper was supported by NSF CNS-0427267, research scientist funds by the Gallaudet Research Institute, NASA Cooperative Agreements 9-58 with the National Space Biomedical Research Institute, CNPq PQ-301278/2004-0, FAEPEX-Unicamp 1679/04, and FAPESP. Carol Neidle provided helpful advice and discussion on the NCSLGR annotations vis-a-vis the tracking results. Lana Cook, Ben Bahan, and Mike Schlang were the subjects in the video sequences discussed in this paper.

References

1. Bauer B, Kraiss K.-F.: Video-based sign recognition using self-organizing subunits. In: International Conference on Pattern Recognition (2002)
2. Blake A., Isard M.: Active Contours : The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion. Springer, Berlin (1999)
3. Blanz V., Vetter T.: A morphable model for the synthesis of 3D faces. In: SIGGRAPH, pp 187–194 (1999)
4. Brandão B., Wainer J., Goldenstein S.: Subspace hierarchical particle filter. In: Brazilian Symposium in Computer Graphics and Image Processing (SIBGRAPI) (2006)
5. Canny J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **8**, 679–698 (1986)
6. Canzler U., Kraiss K.-F.: Person-adaptive facial feature analysis for an advanced wheelchair user-interface. In: Drews, P. (ed.) Conference on Mechatronics and Robotics, vol. 3, pp. 871–876. Sascha Eysoldt Verlag, Aachen (2004)
7. Cootes T.F., Edwards G.J., Taylor C.J.: Active appearance models. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 681–685 (2001)
8. Cootes T.F., Taylor C.J.: Active shape models—their training and application. Comput. Vis. Image Underst. **61**(1), 38–59 (1995)
9. DeCarlo D., Metaxas D., Stone M.: An anthropometric face model using variational techniques. In: Proceedings of the SIGGRAPH, pp. 67–74 (1998)
10. Dimitrijevic M., Ilic S., Fua P.: Accurate face models from uncalibrated and ill-lit video sequences. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 1034–1041 (2004)
11. Ding L., Martinez A.M.: Three-dimensional shape and motion reconstruction for the analysis of american sign language. In: Proceedings of IEEE Workshop on Vision for Human–computer Interaction (V4HCI), (2006)
12. Dinges D.F., Rider R.L., Dorrian J., McGlinchey E.L., Rogers N.L., Cizman Z., Goldenstein S.K., Vogler C., Venkataraman S., Metaxas D.N.: Optical computer recognition of facial expressions associated with stress induced by performance demands. Aviation, Space and Environmental Medicine 76(6 Suppl):B172–B182 (2005)
13. Essa I., Pentland A.: Coding, analysis, interpretation and recognition of facial expressions. IEEE PAMI **19**(7) (1997)
14. Fang G., Gao W., Chen X., Wang C., Ma J.: Signer-independent continuous sign language recognition based on SRN/HMM. In: Wachsmuth I., Sowa T. (eds.) Gesture and Sign Language in Human–computer Interaction. International Gesture Workshop, vol. 2298, pp. 76–85 Lecture Notes in Artificial Intelligence, Springer, Berlin (2001)
15. Fischler M., Bolles R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
16. Goldenstein S.: A gentle introduction to predictive filters. Revista de Informatica Teórica e Aplicada **XI**(1), 61–89 (2004)

17. Goldenstein S., Vogler C., Metaxas D.: Statistical Cue Integration in DAG Deformable Models. IEEE Trans. Pattern Anal. Mach. Intell. **25**(7), 801–813 (2003)
18. Goldenstein S., Vogler C., Metaxas D.: 3D facial tracking from corrupted movie sequences. In: Proceedings of IEEE Computer Vision and Pattern Recognition (2004)
19. Goldenstein S., Vogler C., Velho L.: Adaptive deformable models. In: Proceedings of SIBGRAPI, pp. 380–387 (2004)
20. Goldenstein S., Vogler C.: When occlusions are outliers. In: Proceedings of IEEE Workshop on 25 Years of RANSAC (2006)
21. Kass M., Witkin A., Terzopoulos D.: Snakes: Active Contour Models. Int. J. Comput. Vis. **1**, 321–331 (1988)
22. Metaxas D.: Physics-based Deformable Models: Applications to Computer Vision, Graphics and Medical Imaging. Kluwer Academic Publishers, Dordrecht (1996)
23. Murase H., Nayar S.K.: Visual learning and recognition of 3-d objects from appearance. Int. J. Comput. Vis. **14**, 5–24 (1995)
24. Neidle C., Kegl J., MacLaughlin D., Bahan B., Lee R.G.: The syntax of American Sign Language. Language, Speech, and Communication. MIT, Cambridge (2000)
25. Neidle C., Sclaroff S.: Data collected at the National Center for Sign Language and Gesture Resources, Boston University, under the supervision of C. Neidle and S. Sclaroff. Available online at http://www.bu.edu/asllrp/ncslgr.html, (2002)
26. Pighin F., Szeliski R., Salesin D.: Resynthesizing facial animation through 3D model-based tracking. In: Proceedings of International Conference of Computer Vision, pp 143–150 (1999)
27. Romdhani A., Vetter T.: Efficient, robust and accurate fitting of a 3D morphable model. In: Proceedings of International Conference of Computer Vision, pp 59–66 (2003)
28. Rousseeuw P.J., Van Driessen K.: A fast algorithm for the minimum covariance determinant estimator. Technometrics **41**, 212–223 (1999)
29. Samaras D., Metaxas D., Fua P., Leclerc Y.G.: Variable albedo surface reconstruction from stereo and shape from shading. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp 480–487 (2000)
30. Shi J., Tomasi C.: Good features to track. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp 593–600 (1994)
31. Simoncelli E.: Handbook of Computer Vision and Applications, vol. II, chapter Bayesian Multi-scale Differential Optical Flow, pp. 397–422. Academic, Dublin (1999)
32. Stolfi J., Figueiredo L.: Self-validated Numerical Methods and Applications. 21° Colóquio Brasileiro de Matemática, IMPA (1997)
33. Tao H., Huang T.: Visual estimation and compression of facial motion parameters: elements of a 3D model-based video coding system. Int. J. Comput. Vis. **50**(2), 111–125 (2002)
34. Vogler C., Goldenstein S., Stolfi J., Pavlovic V., Metaxas D.: Outlier rejection in high-dimensional deformable models. Image Vis. Comput. (2006, in press)
35. Vogler C., Metaxas D.: Handshapes and movements: multiple-channel ASL recognition. In: Volpe G., et al. (eds) Proceedings of the Gesture Workshop, vol. 2915 Lecture Notes in Artificial Intelligence, pp 247–258, Springer, Berlin 2004
36. von Agris U., Schneider D., Zieren J., Kraiss K.-F.: Rapid signer adaptation for isolated sign language recognition. In: Proceedings of IEEE Workshop on Vision for Human–computer Interaction (V4HCI) (2006)
37. Wen Z., Huang T.: Capturing subtle facial motions in 3D face tracking. In: Proceedings of International Conference of Computer Vision, pp. 1343–1350 (2003)
38. Zhu Z., Ji Q.: Robust real-time face pose and facial expression recovery. In: Proceedings of IEEE Computer Vision and Pattern Recognition (2006)
39. Zieren J., Kraiss K.-F.: Robust person-independent visual sign language recognition. In: Proceedings of the 2nd Iberian Conference on Pattern Recognition and Image Analysis IbPRIA. Volume Lecture Notes in Computer Science (2005)