A Review on Vision-Based Full DOF Hand Motion Estimation

Ali Erol[(1)] George Bebis[(1)] Mircea Nicolescu[(1)] Richard D. Boyle[(2)] Xander Twombly[(2)]
[(1)]Computer Vision Laboratory, University of Nevada, Reno, NV 89557
[(2)]BioVis Laboratory, NASA Ames Research Center, Moffett Field, CA 94035

Abstract

Direct use of the hand as an input device is an attractive method for providing natural human-computer interaction (HCI). Currently, the only technology that satisfies the advanced requirements of hand-based input for HCI is glove-based sensing. This technology, however, has several drawbacks including that it hinders the ease and naturalness with which the user can interact with the computer controlled environment, and it requires long calibration and setup procedures. Computer vision has the potential to provide much more natural, non-contact solutions. As a result, there have been considerable research efforts to use the hand as an input device for HCI. A very challenging problem in this context, which is the focus of this review, is recovering the 3D pose of the hand and the fingers as glove-based devices do. This paper presents a brief literature review on full degree-of-freedom (DOF) hand motion estimation methods.

1. Introduction

There has been a great emphasis lately in HCI (Human Computer Interaction) research to create easier to use interfaces by making direct use of natural communication and manipulation skills of humans. Except for speech, the direct sensing approach requires motion measurement of various human body parts. The hand, which can technically be seen as a device with more than 20 DOF, forms the most effective, general purpose, interaction tool for HCI. Skill learning systems, surgical simulations, and robot instruction or virtual environments in general are several advanced applications requiring direct sensing of hand and/or finger motion. Common desktop applications such as Computer Aided Design (CAD), film planning, 3D modelling and drawing), have the potential to make use of this technology to enable natural high DOF interaction, which can not be achieved with the conventional Graphical User Interfaces (GUIs).

Currently, the most effective devices for measuring hand motion are electro-mechanical or magnetic sensing devices [41]. These devices are worn on the hand to measure the kinematic parameters (i.e., location of the hand and/or the finger joint angles). However, these devices have several drawbacks in terms of casual use as they are very expensive, hinder the naturalness of hand motion, and require complex calibration and setup procedures to be able to obtain precise measurements. Despite these problems, glove-based input devices deliver the most complete, application independent set of measurements in real-time. The data produced by these devices can be easily processed to extract the articulated motion of the hand and to derive higher level features (e.g., finger tip locations, pointing direction or force generated by a finger) to be interpreted by the application.

Computer vision on the other hand has the potential to provide much more natural, non-contact solutions. Implementations of this idea dates back to late 70s [14] while several real-time system prototypes have been proposed in [25, 28, 34]. A major component of these systems is a gesture recognition engine that can operate without estimating any 3D features of hand motion [30]. Gestures enable the user to issue commands but there is still a need for extracting continuous 3D motion signals to drive dynamic virtual interface elements, which may be as simple as a 3D mouse pointer or as complex as the virtual copy of the hand itself [40]. One solution that provides simple interfaces with real-time operation speed is extracting only the desired 3D hand motion data (fingertip positions, finger orientations and/or global hand pose etc.) without going through a full reconstruction of the hand state. These systems rely extensively on posture and viewpoint restrictions to avoid critical occlusions and keep the appearance of the hand in a reasonable range. A ”point and click” interface where the hand has a ”gun-like” posture is an example of these restrictions. Basic image processing methods that can not be generalized to arbitrary hand poses are used to extract 2D features, which are then mapped to 3D features by fast techniques such as stereo vision.

More recently an alternative approach which aims to recover the full kinematic structure of the hand (see Figure 3), such as in glove-based devices, has received attention. This is a very challenging problem, whose solution is not expected to be very cheap. Since the hand is a flexible object, its projection results in a large variety of shapes with many self-occlusions. Nevertheless, there are several good reasons for tackling this problem. Most importantly, full DOF pose estimation is mandatory for advanced virtual environment applications (e.g. skill learning systems). In the case of simpler desktop interfaces, it is possible to provide

Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)
1063-6919/05 $20.00 © 2005 **IEEE**

principled ways of extracting continuous data by simply fixing the DOF (i.e., restrict hand pose) [31, 37]. It can also be argued that 3D pose data can provide more useful features for gesture recognition purposes as they are view independent and directly related to the hand motion. As an example, hand postures can be easily described using fixed values or intervals of joint angles.

In this paper, we provide a brief review addressing the problem of full DOF hand pose estimation. It should be mentioned that there are several reviews on hand modelling, pose estimation, and gesture recognition [49, 48, 47, 13, 30, 45], the latest of which cover studies up to 2000. However, none of these surveys address the full motion estimation problem in detail as they mainly concentrate on alternative solutions and on the problem of recognizing hand gestures.

It should be mentioned that hand pose estimation has close relationship with human body pose estimation or the pose estimation of articulated objects in general. Human body pose estimation is a more intensive research field. Many algorithms used in hand tracking have a lot of similarities to algorithms proposed previously for human body pose estimation. However, there are also many differences in operation environments and related applications. We have limited the content of this paper to studies directly addressing the problem of hand-pose estimation. A recent survey on human body pose estimation along with pointers to older surveys can be found in [44].

In the next section, we define the problem of full DOF hand pose estimation and provide a categorization of the algorithms that have appeared in the literature. Hand modelling is an important issue to be considered for any model-based method and it is reviewed in Section 3. In Section 4, we consider the problem of feature extraction in the context of hand-pose estimation. In Sections 6, 7, and 5 we discuss in detail methods falling into different categories. In Section 8, we provide a summary of the systems reviewed and their capabilities. Finally, our conclusions are provided in Section 9.

2. Problem and Solutions

The dominant motion observed in hand image sequences is articulated motion; however, there is also some elastic motion but recovering it does not have any major practical use in the majority of applications. Therefore, full DOF hand pose estimation corresponds to estimating the kinematic parameters of the skeleton of the hand. One exception to this common practice is given in [9], where the entire surface of the hand is modelled using principal component analysis (PCA). Such a representation requires further processing to extract useful higher level information such as pointing direction.

There are two main categories of solutions shown in Figure 1: (1) Model based tracking, (2) Single frame pose estimation. The majority of the studies reviewed in this study employ the model-based tracking method, which has been used in many studies for tracking various types of objects in 2D or 3D [23]. A block diagram of a generic system is shown in Figure 2. The framework requires a geometric model of the hand to be constructed off-line. In a general setting, model-based visual pose estimation corresponds to a search in the state space to find the parameters that minimize the matching error between groups of model features and groups of features extracted from the input images. A tracking engine helps to narrow down the search space using a prediction based on the dynamical model of the system. In the first frame, a prediction is not available therefore a separate initial state estimation procedure is required. Most systems solve this problem manually or by assuming a simple known initial configuration (e.g a stretched hand with no occlusion). Prediction is performed by using the history of the hand states and requires identifying a system model. Modelling the non-linear dynamics of the hand motion is not an easy problem [51], therefore, first or second order linear dynamics, that assert smooth state or velocity changes, are assumed. Many systems rely on a local search around the prediction to produce the best estimate at that frame. However, the existence of local minima and discontinuities in the matching error or discontinuities in hand motion does not allow this type of tracker to work well on long sequences [7, 10]. An alternative solution is keeping track of multiple hypotheses, which in principle requires determining all the local maxima in the matching error. These systems implement Bayesian filtering or some approximations using particle filters or grid based methods.

Full DOF Hand Pose Estimation Algorithms		
Single Frame	**Model Based Tracking**	
• Global Search	**Single Hypotheses**	**Multiple Hypotheses**
• Template matching	• Local Search	• Bayesian Filtering
• Inverse kinematics	• Optimization	• Particle Filters
• 2D-3D Mapping	• Force Models	• Grid based filters

Figure 1: Different approaches to hand pose estimation.

Single frame pose estimation on the other hand attacks the problem without making any strong assumptions on time coherence, resulting in a harder problem. However, its solution can lead to algorithms for initialization or re-initialization for tracking based systems. Another motivation for the existence of these systems is the very fast motion capability of the hand and fingers even in a casual manipulation process [40]. The images of consecutive frames can be very different making time coherence assumptions useless. Global search on a large database of templates, inverse kinematic solutions based on fingertip positions, and full bottom up approach where 2D features are mapped directly to the state space are some techniques in this category.

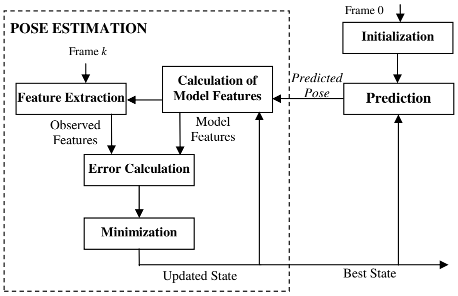

Figure 2: Model based tracking.

3. Hand Modelling

The kinematic model of the hand has more than 20 DOF resulting in a very high dimensional problem. The bones in the skeleton form a system of rigid bodies connected together by joints with one or more degrees of rotation freedom as shown in Figure 3. The pose of the hand is represented by a state vector composed of angular DOFs, which is often called the *local state*, and the six DOFs of the palm frame, which is often called the *global state*. The length of the links between joints are assumed to be fixed and can be estimated separately through a calibration procedure. From an application point of view, capturing the motion of all the bones is not necessary and moreover not very feasible, therefore, several approximations regarding angular DOF of joints are made.

A 27 DOF model that was introduced in [18] and has been used in many studies is shown in Figure 3. The CMC joints are assumed to be fixed, which quite unrealistically asserts that the palm is modelled as a rigid body. The fingers are modelled as serial kinematic chains attached to the palm at anchor points located at MCP joints. The IP, DIP and PIP joints of the fingers are only capable of flexion-extension motion. All five MCP joints have an extra abduction-adduction capability. The TM, which is the main source of flexibility for the thumb, is assumed to be a saddle joint with 2 DOF. Standard robotics techniques provide efficient representations and fast algorithms for various calculations related to the kinematics or dynamics of the model. Adding an extra twist motion to MCP joints [3, 4], introducing one flexion/extension DOF to CMC joints [26] or using a spherical joint for TM [15] are some examples of variations of the kinematic model.

Full DOF hand pose estimation systems extensively rely on a-priori information on the shape of the hand; therefore, the kinematic model is augmented with shape information, which can be obtained using an off-line calibration procedure. However, computational efficiency does not allow the use of very complex shape models. In many studies, the hand model needs to be projected many times on the input image(s) to obtain features that can be compared with observed features. Visibility calculations to avoid occlusions add extra complexity to the projection calculations. These problems have motivated the use of rough shape models composed of simple geometric primitives attached to each link or joint of the hand skeleton. In [37], the primitives used are quadrics. Using projective geometry properties of quadrics, fast algorithms for projecting the quadrics and calculating their visibility are given. In [50], an even more economical view-dependent model called "cardboard model" was presented. When viewed from a direction orthogonal to the palm, the hand is modelled as the union of rectangles attached to each link and the palm. A visibility map is used to handle the visibility calculations. It should be clear that the accuracy of these hand shape representations affect the precision of the pose estimates. Therefore, some studies employ more realistic models. In [15], for example, the skeletal model is covered by a B-spline surface whose control points are attached to the links in the model. In another study [3], a deformable skin model was implemented using computer graphics techniques.

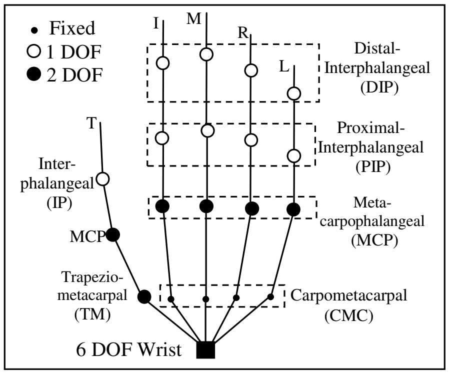

Figure 3: Hand Model

An important issue that can help pose estimation algorithms is the issue of joint angle constraints. Active motion of the hand (i.e., motion without external forces) is highly constrained, which is not reflected in the kinematic model. A first attempt to capture natural hand motion constraints is complementing the kinematic model with *static constraints* that reflect the range of each parameter and *dynamic constraints* that reflect the joint angle dependencies [18, 15]. Based on biomechanics studies, certain closed form constraints can be derived. An important static constraint is the relation $\theta_{DIP} = \frac{2}{3}\theta_{PIP}$ between the PIP and DIP angles that helps to decrease the DOF by 4. Other constraints come technically in the form of inequalities.

The very intricate structure of the hand does not allow expressing all the constraints in a closed form. Moreover, the natural motion of the hand may follow more subtle constraints which have nothing to do with structural limitations [20]. These problems have motivated learning-based approaches which yield hand state representations in much lower dimensional spaces. In [20], PCA was applied to a large amount of joint angle data collected using a glove-based sensor to construct a 7-dimensional space. The data was approximated in the reduced dimensional space as the union of linear manifolds. Maintaining the training samples [22], constructing template databases from these samples [39, 2] or learning the dynamics of hand motion [51] are other means of incorporating the constraints in the search process.

During the pose estimation process, the shape parameters (e.g. lengths of links, size of volumetric primitives) of the hand model are kept fixed and only the joint angles and the global hand position are estimated. The fixed parameters can affect the performance of the model-based estimation method; therefore there is a need for user specific measurement (i.e., hand calibration) of the fixed parameters. This problem is mostly solved manually in the literature. There are only a few studies that elaborate on this problem. In [15], a semi-automatic hand calibration procedure is described. Several landmarks on the hand are found manually in images taken from different views and a spline-based model is fit to the landmark points. In [36], the lengths of links are estimated together with the kinematic model parameters over a sequence. In [5] the image of an open hand and anthropological ratios between finger segments were utilized.

4. Feature Extraction and Matching

The hand creates images that are very difficult to analyze in general. High level features such as fingertips, fingers, joint locations, and the links between joints are are very desirable but also very difficult to extract in a bottom-up manner. The algorithms that require direct extraction of high level features often rely on markers to extract fingertip joint locations or some anchor points on the palm [18, 5, 11]. Assuming a clutter-free background, it is possible to extract some high level features without any markers. [27] uses a markerless fingertip extraction algorithm based on Gabor filters and a special neural network architecture (LLM-net). In [24], contour analysis was performed to detect the intersections of the fingers and the palm.

The majority of studies rely on low-level features that are utilized in matching error calculations during the model fitting phase. The calculation of the matching error requires: (1) extracting a set of features from the input images, (2) projecting the model on the scene (or back-projecting the image features in 3D), and (3) establishing a correspondence between groups of model and image features. The projection of the model features may be computed on the fly or pre-generated and stored in a database. The model features can also be used to drive the feature extraction process. Depending on the amount of occlusion, it is possible to extract high-level features in this case [31].

Contours or edges are somewhat universal features that can be used in any model-based technique [23]. Often, a volumetric model of the hand is projected on the images to calculate the contours of projection. The distance between samples of the contour and the closest edge in the normal direction is used for measuring error [21, 22, 37, 22, 9]. Edge-based features require very simple background to be effective. Distance transforms of edges help to calculate more robust error measures. Chamfer matching was used in several studies [42, 38, 39, 2] demonstrating good performance in cluttered backgrounds. In [38, 42, 39], the edge-based measure was further augmented with color-based measurements. The likelihood of the segmentation asserted by the model projection was calculated using background and skin color models as a measure of similarity. In [2], a probabilistic line matching algorithm was proposed. Combining edges with optical flow and shading assuming uniform static background was proposed in [24]. Silhouette is another frequently used feature [22, 21, 16, 26]. The overlapping area of silhouettes is taken to be a measure of similarity. In [26], the normalized correlation of the distance transform of silhouettes was utilized. Silhouettes can be combined with edge-based measures to increase robustness [21, 22].

Use of 3D features is limited to a few studies. In [6], a stereo camera was used to get a dense 3D reconstruction of the scene, then the hand was segmented by simply thresholding the depth map. The depth map enables dealing with cluttered backgrounds as long as the hand is the closest object to the stereo camera. For each point in the reconstruction, the distance to the closest 3D model point is used as a measure of error. In [3, 4], an active sensor was utilized to get 3D depth data. Skin color was used for segmenting the hand. Points on the hand model surface were paired with points in the reconstruction and the distance between them was used for calculating the error. In [43, 8], the visual hull [17], which is an approximate 3D reconstruction based on silhouettes, was used. A marker-based 3D feature generation method was used in [19, 18] to triangulate the 2D fingertip and palm markers using multiple cameras. A drawback of 3D reconstruction is the additional computational cost. However, 3D information is valuable data that can help eliminate problems due to self-occlusions which are inherent in image-based approaches [6].

5. Single Frame Pose Estimation

Without any hand motion constraints or multiple views, single frame pose estimation problem has multiple solutions, therefore, using these systems over image sequences might not be possible without complementing them with some form of tracking.

One approach is to perform a global search over a database of templates labelled with the pose parameters. In [39], the problem was addressed as an object detection problem using the structure of the tree-based filter given in [42, 38] (see Section 7). The algorithm is equivalent to traversing the tree without any priors to make a classification decision based on the template matching results. In [2], the problem was formulated as an image database indexing problem. Advanced database indexing techniques were utilized for fast chamfer matching and probabilistic line matching over the whole database. For fully unconstrained motion, database search is expected to be very expensive. Both systems demonstrate unconstrained global hand pose estimation that could be used for initialization and gesture recognition purposes.

Fingertip locations are often used for single frame pose estimation. Calculating the joint angles given the fingertip locations is closely related to the classical inverse kinematics problems. Without any motion constraints, however, there exist multiple solutions. Extensive use of hand model constraints helps regularize the problem. For example, the whole finger flexion can be reduced to 1 DOF by relating PIP, DIP and MCP flexion angles [18, 5, 27]. A model fitting procedure, mainly used for hand animation purposes, was proposed in [18]. This system employs markers on the fingertips and the palm and multiple cameras to estimate the locations of these features in 3D. Starting from an arbitrary pose, the skeletal tree is hierarchically updated to reach the extracted fingertip locations. In a more recent marker-based work [5], a single image was used to estimate the pose. Instead of model fitting, closed form solutions were derived to calculate the angles from 2D marker positions under orthographic projection. In [27], a neural network architecture, called PSOM, was trained to construct a mapping from the 2D fingertip positions to joint angles.

In [32], a more general approach was proposed: learning a mapping from a 2D feature space to the state space. The fingertip-based algorithms described above can be seen as a special case of this approach. Instead of fingertips, they used rotation and scale invariant moments of the hand silhouette. The mapping was implemented using a machine learning architecture (Specialized Mapping Architecture (SMA)) without applying any extra hand motion constraints. SMA is capable of generating multiple hypotheses [33].

6. Single Hypothesis Tracking

The most common approach to fitting a model to the extracted features is to use standard optimization techniques. In [31], the error based on joint links and finger tips was minimized using Newton's method augmented with a stabilization algorithm [23]. Stabilization was used to deal with the existence of singularities, which can deteriorate the performance of differential methods in general [7]. In [11], the same technique was applied using fingertip and joint markers. Silhouette-based error measures were minimized using Nelder Mead Simplex (NMS) in [29], and Genetic Algorithms (GAs) and Simulated Annealing (SA) in [26]. Specifically, the NMS algorithm was modified to account for closed form hand model constraints in [29]. In [9], PDM and contour-based edge correspondences were utilized with a weighted least square minimization procedure. The weights were calculated based on the edge strength. In [22], a two stage model fitting algorithm was proposed based on NMS: a coarse stage that constraints the simplex to pass through sample points collected using a glove-based sensor (See Section 3) followed by a fine tuning stage where the simplex can evolve without constraints. Finally, a marker-based system used stereo cameras to extract the 3D locations of a number of markers on the palm and fingertips and applied GAs to estimate the orientation of the palm [19]. The state of the fingers was estimated using inverse kinematics and regression techniques. In a recent system [4], Stochastic Gradient Descent (SGD) along with depth features was proposed. A small number of points on the model surface were selected randomly at each iteration to reduce computational cost and escape local minima. Hand model constraints were carefully taken into consideration by using an additional step at each iteration. The resulting algorithm is called SMD (Stochastic Meta Descent).

An alternative approach to model fitting utilizes physical force models. In this approach, the matching error is used to create forces to be applied on the surface of the articulated model. Then, the motion of the model caused by the forces is calculated and the model parameters are updated. In [6], the forces were derived using the Iterative Closest Point (ICP) algorithm for registering the model with a 3D reconstruction obtained using stereo. As the stereo reconstruction is not a full reconstruction, some modifications were made to the ICP algorithm. Another 3D system described in [43] uses the visual hull of the hand to derive a force model. Using the parts of the model lying outside the visual hull, forces are applied on the link of the skeletal model to push these parts inside the visual hull.

In [46], a 'divide and conquer' approach was proposed. First, the global motion of the hand was estimated, followed by the estimation of the joint angles. This procedure was applied iteratively until convergence. As it is not possible to accurately segment the palm region from the images, outliers are expected. Therefore robust estimation methods were utilized for estimating the pose of the palm. In [21], the ICP algorithm was used assuming orthographic projection. The closest edges to the model contours were used to establish correspondences and a factorization method was applied to calculate the 3 DOF planar global motion. In [21], the NMS algorithm was utilized for estimating the global pose.

Kalman filter has also been used for solving the single hypothesis tracking problem. In [37], the Unscented Kalman Filter (UKF) was used for tracking. UKF applies a deterministic weighted sampling of the Gaussian posterior to be able to track a non-linear system. In [36], the Extended Kalman Filter (EKF) was used where the EKF output was modified by introducing closed form hand motion constraints.

7. Multiple Hypotheses Tracking

The basic idea of multiple hypotheses tracking is to keep multiple pose estimates along the sequence. Keeping multiple hypotheses increases the chances of reaching the true global minimum. This idea is best captured by the Bayesian filtering framework that keeps a probability distribution of the states conditioned on the observations up to the current frame [1]. Many systems in this category provide an approximation to the Bayesian formulation and mainly aim to keep some samples in state space that capture the modes of the posterior.

Particle filtering [1] is a well known technique for implementing a recursive Bayesian filter using Monte Carlo simulations. In [21], importance sampling is utilized for finger pose tracking. The piecewise linear parametrization of the hand configuration space was used (see Section 3) to drive the importance distribution to satisfy this requirement. The success of importance sampling relies on the choice of the importance distribution from which random samples are drawn. The importance distribution should guarantee that the samples drawn have high probability of occurrence (i.e., large weights). They demonstrated that it is possible to track the fingers by keeping an order of magnitude less samples than that of the more conventional condensation algorithm [12]. A problem with particle filters is the requirement on the number of samples to be kept and tested. The most expensive part of a tracking system is the error calculation, therefore, repeating this operation on large amount of samples (i.e., [21] reports using 100 samples) is not desirable.

Semi-parametric particle filters making use of model fitting algorithms provide solutions with less number of samples. The samples representing the modes of the posterior are kept and used to initiate model fitting procedures. In [3], the SMD algorithm was utilized resulting in an 8 particle tracker while [22] uses the two stage NMS algorithm with 30 particles.

Another approach to implementing the Bayesian tracking is grid-based filtering [1]. A grid-based approach is followed in [38] by partitioning the state space using a regular multi-resolution grid. Then, the posterior is approximated to be piecewise constant over the nodes of the multi-resolution tree. For each node of the tree, a template, generated using an artificial hand model, is stored. During tracking, the tree is traversed in a depth-first order to update the probabilities. However, it is possible to skip children nodes of low resolution having low probability masses. The tree is constructed using a piecewise linear function that approximates the hand motion data collected using a glove-based device. In a later study [42], alternative tree construction methods were proposed and the hand dynamics were captured by keeping a histogram of the tree node transitions in the training data.

Another study that follows a template matching approach is given in [35]. Silhouette contour features and some scale and rotation invariant features were generated using an artificial hand model and stored in a database. During tracking, several hypotheses were kept and the neighborhood around each hypothesis was searched to find the best matching template and establish new hypotheses. Once the best match from the database had been found, it was further refined using a model fitting algorithm.

8. Summary and Discussion

The key characteristics of the hand pose estimation systems reviewed in this study are summarized in Table 1. The first column provides the reference number while the other columns provide the key characteristics of each system. Specifically, we report: 1) the effective number of DOF that the system targets (i.e., the final DOF after possible degree reduction due to constraints), 2) the number and type of cameras used, 3) the ability of the system to operate in a cluttered background, 4) the features used, 5) the approach used to enforce hand model constraints, 6) the type of the system according to the taxonomy used in this study, 7) systems using a database of templates, 8) details of the algorithm, and 9) observed restrictions in the experimental evaluation.

One of the key issues in evaluating system performance is the availability of ground-truth data. Obtaining ground truth data for 3D hand pose estimation is a difficult problem. Some studies report results on synthetic data [26, 32, 21, 43], while others project the hand model on the input image(s) to show how well the projection(s) match the image data. If we discard very limited motions, the most common assumption when testing a system is keeping the palm parallel to the camera (i.e., facing the camera). The main reason for this assumption is to avoid self-occlusions which are difficult to handle using a single camera. This is confirmed in [32] where it was shown that the pose estimation error of their system is increasing with palm rotation.

It is worth mentioning that among the studies reviewed, there are only two real-time systems. The first of them is the rather old DigitEyes system [31]. It works at 10Hz on an image processing board and can track 3 fingers in 5 DOF motion and the hand in 3 DOF planar motion. The second is the template matching system given in [35]. It is implemented using a PC cluster consisting of 6 PCs and operates at 30Hz. Some tracking results under severe occlusion have been demonstrated using this system.

9. Conclusions

In this paper, we have reviewed a number of studies addressing the problem of full hand motion estimation. The existence of an expensive but high speed system is quite encouraging [35]. However, the lack of an implementation that is part of a real world system indicates that there is still a lot of open theoretical questions. Model-based vision seems to have good potential; however, it has two main deficiencies: (1) high processing requirements, (2) lack of automatic model calibration algorithms. These systems do not have the potential to run real-time on today's desktop computers but, on the other hand, they do have a place in advanced virtual environment applications, which can afford more expensive systems.

The general trend in these systems is building single camera systems. However, there is also an inevitable tendency to avoid occlusions by keeping the global hand pose fixed with respect to the camera. Multiple camera systems and 3D features are not explored very well. Although these systems are more expensive, they can provide better ways to handle occlusions and can lead to more accurate hand tracking systems for advanced tasks such as virtual object manipulation.

Acknowledgement: This work was supported by NASA under grant NCC5-583

References

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):173–188, February 2002.

[2] V. Athitsos and S. Sclaroff. Estimating 3d hand pose from a cluttered image. In *Conference on Computer Vision and Pattern Recognition (CVPR '03)*, volume I, Madison, Wisconsin, 2003.

[3] M. Bray, E. Koller-Meier, and L. V. Gool. Smart particle filtering for 3d hand tracking. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR04)*, 2004.

[4] M. Bray, E. Koller-Meier, P. Mueller, L. V. Gool, and N. N. Schraudolph. 3d hand tracking by rapid stochastic gradient descent using a skinning model. In *1st European Conference on Visual Media Production (CVMP) London*, 2004.

[5] C. S. Chua, H. Y. Guan, and Y. K. Ho. Model-based finger posture estimation. In *ACCV2000*, 2000.

[6] Q. Delamarre and O. Faugeras. 3d articulated models and multiview tracking with physical forces. *Comput. Vis. Image Underst.*, 81(3):328–357, 2001.

[7] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Computer Vision and Pattern Recognition*, volume 2, pages 126–133. IEEE, 2000.

[8] A. Erol, G. Bebis, R. Boyle, and M. Nicolescu. Visual hull construction using adaptive sampling. In *IEEE Workshop on Applications of Computer Vision*, 2005.

[9] T. Heap and D. Hogg. Towards 3d hand tracking using a deformable model. In *2nd International Conference on Face and Gesture Recognition 96*, 1996.

[10] T. Heap and D. Hogg. Wormholes in shape space: tracking through discontinuous changes in shape. In *Sixth International Conference on Computer Vision*, pages 344–349, Bombay , India, 1998.

[11] E. Holden. *Visual Recognition of Hand Motion*. PhD thesis, Department of Computer Science, University of Western Australia, 1997.

[12] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference on Computer Vision*, volume 1, pages 343–356, Cambridge UK, 1996.

[13] M. Kohler and S. Schroter. A survey of video-based gesture recognition - stereo and mono systems. Technical Report Nr. 693/1998, Informatik VII, University of Dortmund, August 1998.

[14] M. W. Krueger, T. Gionfriddo, and K. Hinrichsen. Videoplacean artificial reality. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 35–40. ACM Press, 1985.

[15] J. J. Kuch and T. S. Huang. Human computer interaction via the human hand: A hand model. In *Twenty-Eighty Asilomar Conference on Signal, Systems, and Computers*, pages 1252– 1256, 1994.

[16] J. J. Kuch and T. S. Huang. Vision based hand modeling and tracking for virtual teleconferencing and telecollaboration. In *Fifth International Conference on Computer Vision*, June 1995.

[17] A. Laurentini. The visual hull concept for silhouette-based image understanding. *PAMI*, 16(2), 1994.

[18] J. Lee and T. Kunii. Constraint-based hand animation. In *Models and Techniques in Computer Animation*, pages 110–127. Springer, Tokyo, 1993.

[19] C. C. Lien and C. L. Huang. Model based articulated tracking for gesture recognition. *Image and Vision Computing*, 16:121–134, 1998.

[20] J. Lin, Y. Wu, and T. S. Huang. Modeling the constraints of human hand motion. In *IEEE Human Motion Workshop*, pages 121–126, 2000.

[21] J. Lin, Y. Wu, and T. S. Huang. Capturing human hand motion in image sequences. In *Workshop on Motion and Video Computing (WMVC02)*, pages 99–104, 2002.

[22] J. Y. Lin, Y. Wu, and T. S. Huang. 3d model-based hand tracking using stochastic direct search method. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 693+, Seoul, Korea, May 2004.

[23] D. G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441–450, 1991.

[24] S. Lu, G. Huang, D. Samaras, and D. Metaxas. Model-based integration of visual cues for hand tracking. In *WMVC 2002*, 2002.

[25] C. Maggioni and K. B. Gesturecomputer - history, design and applications. In R. Cipolla and A. Pentland, editors, *Computer Vision for Human-Machine Interaction*, pages pp. 312–325. Cambridge, 1998.

[26] K. Nirei, H. Saito, M. Mochimaru, and S. Ozawa. Human hand tracking from binocular image sequences. In *22th International Conference on Industrial Electronics, Control, and Instrumentation*, 1996.

[27] C. Nölker and H. Ritter. Grefit: Visual recognition of hand postures. In A. Braffort, R. Gherbi, S. Gibet, J. Richardson, and D. Teil, editors, *Gesture-Based Communication in Human-Computer Interaction: Proc. International Gesture Workshop, GW 99, France*, pages 61–72. Springer Verlag, LNAI 1739, 1999.

[28] R. G. O'Hagan, A. Zelinsky, and S. Rougeaux. Visual gesture interfaces for virtual environments. *Interacting with Computers*, 14:231–250, 2002.

Table 1: Summary of the systems reviewed. Abbreviations: C→closed form constraints, L→learning-based constraints, SH→Single Hypothesis, MH→Multiple hypotheses, SF→single frame, Y→Yes. The entries outside the parentheses refer to the pose of the fingers while the entries within parentheses refer to the global pose.

Ref.	DOF	Camera	Clut.	Features	Constr.	Method	Templ.	Details	Restrictions
[4]	20(6)	1 Depth	Y	Depth	C	SH		Stochastic Gradient Descent	
[6]	21(0)	1 Depth	Y	Depth		SH		Force Model	Palm faces camera
[9]	N/A	1	Y	Edge	L	SH		Weighted Least Squares	
[11]	15(6)	1		Marker		SH		Gauss Newton	Palm faces camera
[19]	17(6)	2		Marker	C	SH		Inverse kinematics(GA)	Palm faces camera
[24]	20(6)	1		Edge, Opt. Flow, Silh.,Fingers		SH		Force model	
[26]	27(6)	2		Silh., Opt. Flow		SH		NMS,GA	No global motion
[29]	Unclear(6)	1		Edge, Silh.	C	SH		NMS	Limited finger motion
[31]	21(6)	1		Finger tip-link		SH		Gauss Newton	5(3) DOF Tracking
[37]	21(6)	1		Edge		SH		UKF	1(6) DOF tracking
[43]	21(6)	4		Visual Hull		SH		Force model	Limited finger, no global
[3]	20(6)	1 Depth	Y	Depth	C	MH		Particle Filter	
[35]	Unclear(0)	1		Contours, moments		MH	Y	Template matching	
[36]	20(6)	1		Silh., finger tip,link	C	MH		EKF	
[38]	21(6)	1	Y	Edge, Color	L	MH	Y	Tree-Based Filter	
[42]	20(0)	1	Y	Edge, Color	L	MH	Y	Tree-Based Filter	Palm faces camera
[21]	20(3)	1		Edge, Silh.	L	MH(SH)		Particle Filter(ICP)	Palm faces camera
[22]	21(6)	1	Y	Edge, Silh.	L	MH(SH)		Particle Filter(NMS)	
[2]	N/A	1	Y	Edge,lines		SF	Y	Database Indexing	0(6) DOF Initialization
[5]	6(6)	1		Marker		SF		Inverse Kinematics	
[18]	14(6)	2		Marker	C	SF		Model Fitting	
[27]	10(0)	1		Finger tip	C	SF		Inverse Kinematics	Palm faces camera
[32]	22(0)	1	Y	Moments		SF		2D-3D Mapping	
[39]	N/A	1	Y	Edge, Color		SF		Tree Based Filter	0(6) DOF initialization

[29] H. Ouhaddi and P. Horain. 3d hand gesture tracking by model registration. In *Int. Workshop on Synthetic - Natural Hybrid Coding and Three Dimensional Imaging (IWSNHC3DI'99*, 1999.

[30] V. I. Pavlovic, R. S. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *PAMI*, 1997.

[31] J. Rehg and T. Kanade. Digiteyes: Vision-based hand tracking for human-computer interaction. In *Workshop on Motion of Non-Rigid and Articulated Bodies*, pages 16–24, November 1994.

[32] R. Rosales, V. Athitsos, L. Sigal, and S. Sclaroff. 3d hand pose reconstruction using specialized mappings. In *International Conference on Computer Vision (ICCV01)*, volume 1, pages 378–385, 2001.

[33] R. Rosales and S. Sclaroff. Algorithms for inference in specialized maps for recovering 3d hand pose. In *Fifth IEEE International Conference on Automatic Face and Gesture Recognition (FGR02)*, 2002.

[34] J. Segen and S. Kumar. Gesture vr: vision-based 3d hand interace for spatial interaction. In *6th ACM international conference on Multimedia*, 1998.

[35] N. Shimada, K. Kimura, and Y. Shirai. Real-time 3d hand posture estimation based on 2d appearance retrieval using monocular camera. In *ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 23–30, Vancouver, BC, Canada, 2001. IEEE.

[36] N. Shimada, Y. Shirai, Y. Kuno, and J. Miura. Hand gesture estimation and model refinement using monocular camera - ambiguity limitation by inequality constraints. In *3rd IEEE Int'l Conf. on Face and Gesture Recognition*, 1998.

[37] B. Stenger, P. R. S. Mendonca, and R. Cipolla. Model-based 3d tracking of an articulated hand. In *CVPR 2001*, 2001.

[38] B. Stenger, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Filtering using a tree-based estimator. In *ICCV 2003*, October 2003.

[39] B. Stenger, A. Thayananthan, P. H. S. Torr, and R. Cipolla. Hand pose estimation using hierarchical detection. In *Intl. Workshop on Human-Computer Interaction*, 2004.

[40] D. J. Sturman. *Whole Hand Input*. PhD thesis, MIT, February 1992.

[41] D. J. Sturman and D. Zeltzer. A survey of glove-based input. *IEEE Comput. Graph. Appl.*, 14(1):30–39, 1994.

[42] A. Thayananthan, B. Stenger, P. H. S. Torr, and R. Cipolla. Learning a kinematic prior for tree-based filtering. In *Proc. British Machine Vision Conference*, 2003.

[43] E. Ueda, Y. Matsumoto, M. Imai, and T. Ogasawara. A hand-pose estimation for vision-based human interfaces. *IEEE Transactions on Industrial Electronics*, 50(4), August 2003.

[44] L. Wang, W. Hu, and T. Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, March 2003.

[45] R. Watson. A survey of gesture recognition techniques. Technical Report TCD-CS-93-11, Trinity College, Dublin 2, 1993.

[46] Y. Wu and T. S. Huang. Capturing articulated human hand motion: A divide and comquer approach. In *Int'l Conf on Computer Vision*, 1999.

[47] Y. Wu and T. S. Huang. Human hand modeling and animation in the context of hci. In *Int. Conf. On Image Processing*, 1999.

[48] Y. Wu and T. S. Huang. Vision-based gesture recognition: A review. *Lecture Notes in Computer Science*, 1739:103+, 1999.

[49] Y. Wu and T. S. Huang. Hand modeling analysis, and recognition. *IEEE Signal Processing Magazine*, 18(3), May 2001.

[50] Y. Wu, Y. L. Lin, and T. S. Huang. Capturing natural hand articulation. In *IEEE Int'l Conf. on Computer Vision (ICCV01)*, pages 426–432, 2001.

[51] H. Zhou and T. S. Huang. Tracking articulated hand motion with eigen dynamics analysis. In *Ninth IEEE International Conference on Computer Vision*, 2003.