Capturing Human Hand Motion in Image Sequences

John Lin[†], Ying Wu[‡], and Thomas S. Huang[†]

[†] University of Illinois at Urbana-Champaign, 405 N. Mathews, Urbana, IL 61801
[‡] Northwestern University, 2145 Sheridan Road, Evanston, IL 60208-3118
[†]{jy-lin, huang}@ifp.uiuc.edu, [‡]yingwu@ece.nwu.edu

Abstract

Visually capturing human hand motion requires estimating the 3D hand global pose as well as its local finger articulations. This is a challenging task that requires a search in a high dimensional space due to the high degrees of freedom that fingers exhibit and the self occlusions caused by global hand motion. In this paper we propose a divide and conquer approach to estimate both global and local hand motion. By looking into the palm and extra feature points provided by fingers, the hand pose is determined from the palm using Iterative Closed Point (ICP) algorithm and factorization method. The hand global pose serves as the base frame for the finger motion capturing. Noticing the natural hand motion constraints, we propose an efficient tracking algorithm based on sequential Monte Carlo technique for tracking finger motion. To enhance the accuracy, pose estimations and finger articulation tracking are performed in an iterative manner. Our experiments show that our approach is accurate and robust for natural hand movements.

1. Introduction

Hand gestures can be a more natural and articulate way for many human computer interaction applications. For instance, people can use their hands to manipulate virtual objects directly in virtual environments. But one of the main difficulties is how to capture human hand motion. As an alternative to glove-based techniques that requires users to wear a special data glove, vision-based techniques are non-invasive and more affordable. However, capturing hand and finger motion from video sequences is an extremely challenging task. Typical hand motions consist of global translation, rotation, and natural finger movements. Due to the hand anatomy, hand rotations often introduce self-occlusions that cause some of the fingers to become invisible. Since the finger motion has high degrees of freedom, many techniques of estimating finger articulations often involve a formidable search problem in a high dimensional space.

Different methods have been proposed to analyze human hand motion for visual hand tracking. One approach makes use of deformable hand shape models [2], in which the hand shape deformation could be governed by Newtonian dynamics or statistical training methods such as the Principal Component Analysis (PCA). However, accurate estimates of hand poses are hard to obtain by these methods.

Another approach is the appearance-based approach, which tries to establish the mapping between image feature space and the hand motion space [7, 13]. However, the mapping can be difficult to learn and may not be one-to-one. Also, it is not trivial to collect large and representative set of training data.

A third approach is the 3D model-based approach. The hand motion could be estimated by matching the 3D model projections and observed image features, so that the problem becomes a search problem in a high dimensional space. To construct the correspondences between the model and the images, different image observations have been studied, such as fingertips [5, 9, 12], line features [6, 8], contours [10], and silhouettes [4, 1, 14].

Many methods tackle the global hand motion and local finger motion simultaneously, such that the optimization would have a very high chance to converge to a local minima. On the other hand, a divide-and-conquer approach [12] could be taken to separate the hand pose determination and articulation estimation. It could be a general method for articulate objects.

This paper proposes an approach to capture both hand pose and finger articulation in the divide-and-conquer framework. We propose an algorithm combining the Iterative Closed Point (ICP) algorithm and the factorization method to determinate global hand pose. Section 3 will describe this approach. The fact that the natural finger motion is also highly constrained helps us reduce the dimensionality of the feasible hand configuration space. We propose to

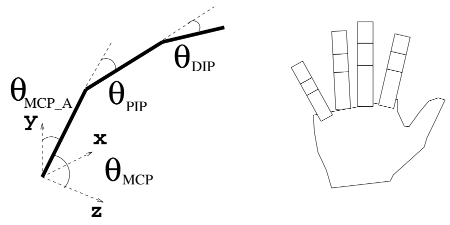
0-7695-1860-5/02 $17.00 © 2002 IEEE

use a set of linear manifolds to characterize hand configuration. Details of the hand motion model could be found in Section 2. In Section 4, an efficient algorithm based on sequential Monte Carlo techniques for tracking finger motion is given. To enhance the accuracy, pose estimations and finger articulation tracking are performed in an iterative manner. Details will be given in Section 5. Our experiments including simulation and real hand sequences will be shown in Section 6.

2. Hand Model and Motion Constraints

Hand motion consists of global hand pose M_G and local finger articulation M_L. Global hand motion can be described by 3D translation t and rotation $\mathbf{R}$ of the palm. The local finger motion is represented by a set of joint angles Θ. The hand structure is represented by a kinematical model (Figure 1a), which has roughly 20 degrees of freedom [5, 4]. The task of motion capturing is to estimate $\{\mathbf{R}, t, \Theta\}$.

In our experiment, we use a cardboard model in which each finger is represented by a set of three connected planar patches. The parameters of the patches are calibrated according to an individual user. (Figure 1b). Although it is a simplification of the human hand structure, it offers a good approximation for motion capturing. A major drawback is that it cannot capture motions with large out-of-plane rotations.



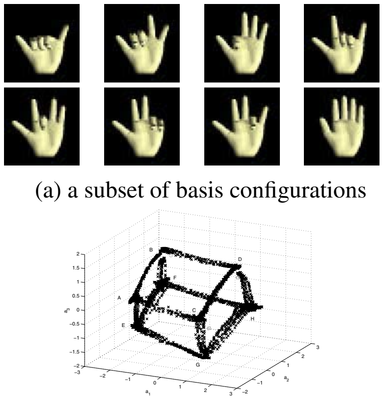
(a) (b)

Figure 1. Hand Model: (a) Kinematical chain of one finger, (b) Cardboard hand model.

Instead of searching in the 20 dimensional space, we would like to use various constraints to reduce the dimensionality of the joint angle space and find a smaller feasible space, which we will call the configuration space Ξ. Several commonly known constraints due to the anatomy of the hand can be used to initially reduce the dimensionality to roughly 15. To further reduce the dimensionality, we have collected more than 30,000 joint angle data from various hand motions using CyberGlove. Then PCA is applied to eliminate the redundancy. We can project $\Theta \in \mathcal{R}^{20}$ into a 7-dimensional subspace while maintaining 95% of the information. Therefore, the configuration space Ξ is defined in $\mathcal{R}^7$. Furthermore, we define 28 basis configurations as follows. For each basis state $\mathbf{b}_i$, each finger is either fully extended or fully curled. A subset of the basis states is shown in Figure 2a. Our observations of the motion trajectories between basis states in Ξ show that they are roughly linear and that natural hand articulation can be characterized by these linear manifold $\mathcal{L}_{ij}$ spanned by $\mathbf{b}_i$ and $\mathbf{b}_j$, with $i \neq j$.

$$\Xi \approx \bigcup_{i,j} \mathcal{L}_{ij}, \text{ where } \mathcal{L}_{ij} = span(\mathbf{b}_i, \mathbf{b}_j) \qquad (1)$$

A lower dimensional illustration is shown in Figure 2b, in which each point represents a real hand configuration in Ξ.



(a) a subset of basis configurations

(b) linear manifolds in the configuration space

Figure 2. Hand articulation in the configuration space, which is characterized by a set of basis configurations and linear manifolds.

3. Capturing Global Motion

The global hand motion is defined by the current pose estimate of the palm, which is approximated as a rigid planar object, relative to the initial pose. In this section, we present algorithms for determining the pose and estimating the global motion.

3.1. Pose Estimation

In this section, we assume the correspondences have been constructed for pose determination. The process of building the correspondences will be presented in Section 3.2. Let a point on the plane be $\mathbf{x}_i = [x_i, y_i]^T$, and its image point be $\mathbf{m}_i = [u_i, v_i]^T$. Under scaled orthographic projection, we can write $t_3\mathbf{m}_i = \mathbf{A}\mathbf{x}_i + \mathbf{t}$, where

$$\mathbf{A} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \text{ and } \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

We can subtract the centroid of the projection points and model points, i.e., $\hat{\mathbf{m}}_i = \mathbf{m}_i - \bar{\mathbf{m}}$ and $\hat{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}$, which gives $t_3\hat{\mathbf{m}}_i = \mathbf{A}\hat{\mathbf{x}}_i$. Letting $\mathbf{B} = \mathbf{A}/t_3$, we have $\hat{\mathbf{m}}_i = \mathbf{B}\hat{\mathbf{x}}_i$.

Denoting $[u_i^k, v_i^k]^T$ to be the i-th image point at the k-th frame, we define

$$\mathbf{W} = \begin{bmatrix} u_1^1 & u_2^1 & \cdots & u_N^1 \\ v_1^1 & v_2^1 & \cdots & v_N^1 \\ u_1^2 & u_2^2 & \cdots & u_N^2 \\ v_1^2 & v_2^2 & \cdots & v_N^2 \end{bmatrix} = \mathbf{MS}$$

where

$$\mathbf{M} = \begin{bmatrix} M^1 \\ M^2 \end{bmatrix} \text{ and } \mathbf{S} = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \\ y_1 & y_2 & \cdots & y_N \end{bmatrix}$$

The factorization method [11] can be used to solve for $\mathbf{M}$ and $\mathbf{S}$ up to a matrix $\mathbf{D}$, which could be determined by the constraints of $\mathbf{M}$.

After recovering $\mathbf{M}$, which contains $\mathbf{B}$ and t_3, $\mathbf{R}$ and t_3 are computed from $\mathbf{B}$ (see Appendix), which leads to :

$$\begin{bmatrix} t_1 \\ t_2 \end{bmatrix} = t_3\bar{\mathbf{m}} - \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \bar{\mathbf{x}}$$

For simplicity, we can use the first frame that shows the front of the palm for initialization and calibration, and take image points along the palm contour as the model points.

3.2. Iterative Closed Points

The pose determination method presented in the previous section assumes point correspondences. In this section we describe a method for establishing point correspondences by adapting the idea of the Iterative Closed Point (ICP) algorithm. A comprehensive description of ICP for free-form curve registration can be found in [15]. The basic idea is to refine the correspondences and the motion parameters iteratively.

Since we treat the palm as a rigid planar object, it can be represented by its contour curve. A curve can be represented by a set of points. Let $\mathbf{x}_j (1 \le j \le N)$ be the N chained points on the 3D curve model $\mathcal{C}$. Let $\mathcal{C}'$ be the edges observed in the image. The objective is to construct the correspondences between the two curves, such that

$$\mathcal{F}(\mathbf{R}, \mathbf{t}) = \sum_{j=1}^{N} w_j D(\mathbf{P}(\mathbf{R}\mathbf{x}_j^t + \mathbf{t}), \mathcal{C}') \qquad (2)$$

is minimized, where $D(\mathbf{x}, \mathcal{C}')$ denotes the distance of the point $\mathbf{x}$ and the curve $\mathcal{C}'$, w_j takes value 1 if there is a match for $\mathbf{x}_j$ and takes 0 otherwise, and $\mathbf{P}(\mathbf{x})$ represents the projection of $\mathbf{x}$ from 3D space onto the image plane.

The ICP algorithm takes the image edge point that is closest to the projected 3D model point i.e., $\mathbf{P}(\mathbf{R}\mathbf{x}_k^t + \mathbf{t})$,as its correspondence. If all image edge points are too far from the projection, this model point $\mathbf{x}_k$ can not be matched and we set $w_k = 0$. Motion $(\mathbf{R}, \mathbf{t})$ is computed based on such a temporary correspondences using the pose determination method present in Section 3.1. The computed motion results in a new matching. Iteratively applying this procedure, ICP yields a better and better pose estimation, provided that the motion between successive frames are small. It should be pointed out that the ICP procedure converges only to a local minima, which means that we need a close initial start. Obviously, the ICP algorithm could be easily extended to two-frame registration.

4. Capturing Finger Articulation

In this section, we present a sequential Monte Carlo algorithm that takes advantage of the natural hand motion constraints in the tracking algorithm.

4.1. Sequential Monte Carlo

The tracking problem is formulated as a process of conditional probability density propagation. We can track the finger motions efficiently using sequential Monte Carlo method, which offers a way to approximate the evolution of the densities. Denote the target state and image observations by $\mathbf{X}_t$ and $\mathbf{Z}_t$ respectively, and $\underline{Z}_t = \{\mathbf{Z}_1, \ldots, \mathbf{Z}_t\}$, the tracking problem is formulated as:

$$p(\mathbf{X}_{t+1}|\underline{Z}_{t+1}) \propto p(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1}) p(\mathbf{X}_{t+1}|\underline{Z}_t) \qquad (3)$$

The posteriori $p(\mathbf{X}_t|\underline{Z}_t)$ is represented by a set of random samples $\{s_t^{(n)}, \pi_t^{(n)}\}$ which evolve to a new set of samples $\{s_{t+1}^{(n)}, \pi_{t+1}^{(n)}\}$ at time $t+1$ to represent the new posteriori. Different sampling schemes can be used depending on the source of sampling priors [14, 3].

Since finger articulation involves a high DOF, algorithms such as CONDENSATION will require a large number of samples for representing the density propagation, and an intensive computation will be unavoidable. Fortunately, we may reduce the complexity by making use of the finger motion constraints as an outside prior for the importance sampling technique. Let $f_t(\mathbf{X}_t^{(n)}) = p(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\underline{Z}_{t-1})$ be the tracking prior. To approximate the posterior $p(\mathbf{X}_t|\underline{Z})$, we draw random samples from another distribution $g_t(\mathbf{X}_t)$, instead of the prior density $f_t(\mathbf{X}_t)$. Below we will give a brief description of this method. The details can be found in [14].

For natural hand motion, each hand configuration $\mathbf{X}$ should be either around a basis state $\mathbf{b}_k, k = 1, \ldots, M$, or on the manifold $\mathcal{L}_{ij}$, where $i \ne j, i, j = 1, \ldots, M$. Suppose at time frame t, the hand configuration is $\mathbf{X}_t$. We find the projection $\bar{\mathbf{X}}_t$ of $\mathbf{X}_t$ onto the nearest manifold $\mathcal{L}_{ij}^*$, and obtain

$$s_t = 1 - \frac{(\mathbf{X}_t - \mathbf{b}_i)^T(\mathbf{b}_j - \mathbf{b}_i)}{||(\mathbf{b}_j - \mathbf{b}_i)||}$$

Then, random samples are drawn from the manifold $\mathcal{L}_{ij}$ according to the density p_{ij}, i.e.,

$$\begin{aligned} s_{t+1}^{(n)} &\sim p_{ij} = N(s_t, \sigma) & (4) \\ \widetilde{\mathbf{X}}_{t+1}^{(n)} &= s_{t+1}^{(n)}\mathbf{b}_i + (1 - s_{t+1}^{(n)})\mathbf{b}_j & (5) \end{aligned}$$

Next, perform random walk on $\widetilde{\mathbf{X}}_{t+1}^{(n)}$ to obtain hypothesis $\mathbf{X}_{t+1}^{(n)}$, i.e.,

$$\mathbf{X}_{t+1}^{(n)} \sim N(\widetilde{\mathbf{X}}_{t+1}^{(n)}, \Sigma_{t+1}) \qquad (6)$$

Define the importance function to be: $g_{t+1}(\mathbf{X}_{t+1}^{(n)}) = p(s_{t+1}^{(n)}|s_t)p(\mathbf{X}_{t+1}^{(n)}|\widetilde{\mathbf{X}}_{t+1}^{(n)})$, and we have

$$\begin{aligned} g_{t+1}(\mathbf{X}_{t+1}^{(n)}) &\sim \frac{1}{\sigma|\Sigma|^{1/2}} \exp\{-\frac{(s_{t+1}^{(n)} - s_t)^2}{2\sigma^2} \\ &- \frac{1}{2}(\mathbf{X}_{t+1}^{(n)} - \widetilde{\mathbf{X}}_{t+1}^{(n)})\Sigma^{-1}(\mathbf{X}_{t+1}^{(n)} - \widetilde{\mathbf{X}}_{t+1}^{(n)})\} \end{aligned}$$

Finally, the weights must be properly compensated:

$$\pi_{t+1}^{(n)} = \frac{f_{t+1}(\mathbf{X}_{t+1}^{(n)})}{g_{t+1}(\mathbf{X}_{t+1}^{(n)})} p(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1} = \mathbf{X}_{t+1}^{(n)}) \qquad (7)$$

If the previous hand configuration is at one of the basis configurations, say $\mathbf{X}_t = \mathbf{b}_k$, it is reasonable to assume that it selects any one of the manifolds of $\{\mathcal{L}_{kj}, j = 1, \ldots, M\}$ with the same probability. Consequently, random samples are drawn from a mixture density p_k.

4.2. Model Matching

We employ edge observations to measure the likelihood of hypotheses, i.e., $p(\mathbf{Z}_t|\mathbf{X}_t)$ as in [14]. Self-occlusion is handled by constructing an occlusion map for the hand model. The cardboard model for the hand is sampled at a set of K points on the laterals of the patches. For each of these samples, edge detection is performed on the points along the normal of this sample. If we assume that M edge points $\{z_m, m = 1, \ldots, M\}$ are observed, and the clutter is a Poisson process with density λ, then,

$$p_k^e(\mathbf{z}|x_k) \propto 1 + \frac{1}{\sqrt{2\pi}\sigma_e q\lambda} \sum_{m=1}^{M} \exp - \frac{(z_m - x_k)^2}{2\sigma_e^2}$$

We also consider the silhouette measurements, by calculating the difference between the areas of the image A_I and the projected cardboard model A_M, i.e., $p^a \propto \exp - \frac{(A_I - A_M)^2}{2\sigma_a^2}$. Thus, the likelihood becomes:

$$p(\mathbf{Z}|\mathbf{X}) \propto p^a \prod_{k=1}^{K} p_k^e \qquad (8)$$

5. Divide and Conquer Approach

Rather than estimating both global and local motion altogether, another approach is to estimate global and local motions separately and combine the results in an iterative manner [12]. By estimating the global and local parameters separately, we have the advantage of reduced complexity. However, the pose estimation method in Section 3 would induce inaccuracies, since the method matches the palm against all the edges observed in the images. Part of the index and little finger edge points will often be confused as palm edge points, which would result in wrong scaling and rotation estimates.

One remedy is to introduce additional feature points such as the fingertip locations to help reducing the error and provide bounds for the model matching. Our experiments showed that this indeed improves the performance of estimation. We combine the results of global and local parameters in a two-step iterative manner: (1) pose determination based on palm contour and some extra points, using the method describe in Section 3; (2) tracking local finger configurations using a Monte Carlo based algorithm as described in Section 4 and finding extra feature points. The iterations between global and local hand motion estimation would converge to a local stationary point that minimizes the discrepancies between the image observation and model projection. A proof of the convergence can be found in [12].

6. Experiment

We have performed several simulation experiments to evaluate the proposed algorithm, and applied our algorithm to real hand image sequences.

6.1. Simulation

It is generally difficult to obtain the ground truth of hand motions from a real video sequence. Therefore, we have produced a synthetic sequence of 200 frames containing typical hand movements. This synthetic sequence will facilitate the quantitative evaluations of our algorithm. Some examples of the synthetic motion is shown in Figure 3.

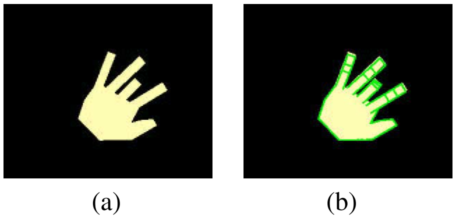


Figure 3. Sample of our results on synthetic sequences. (a) a synthetic image (b) the image with model aligned.

Some of the motion parameters are shown in Figure 4 for comparison. The solid curves are our estimates and the dash curves are the ground truth. The figure plot the x translation with average error of 3.98 pixels, the rotation with average error of 3.42^o, the PIP joint of the index finger with average error of 8.46^o, the MCP flexion of the middle finger with average error of 4.96^o, the PIP joint of the ring finger with average error of 5.79^o, and the MCP abduction of the ring finger with average error of 1.52^o.

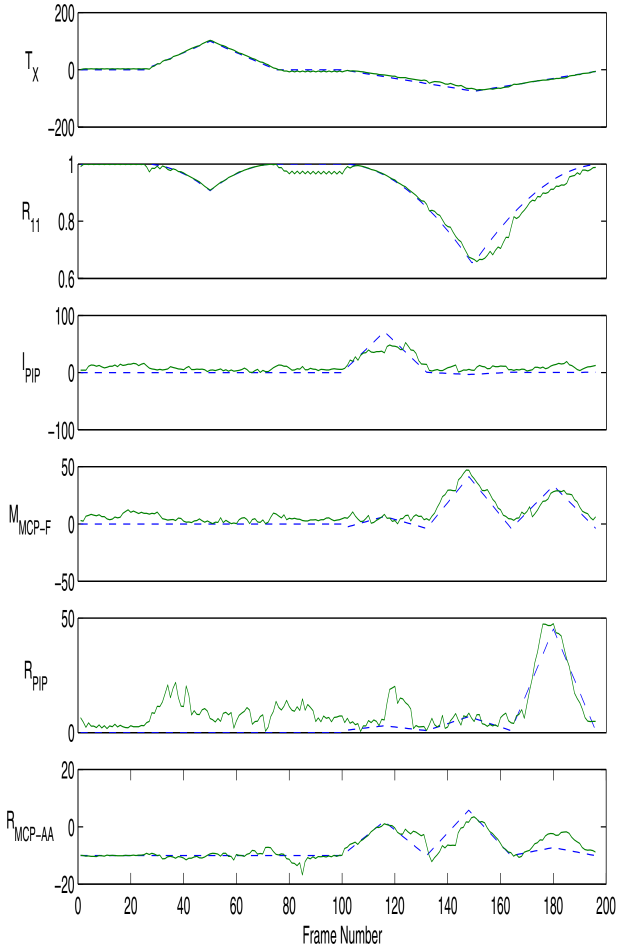


Figure 4. The comparison of our results and the ground truth on a synthetic sequence. The dash curves are the ground truth, and the solid curves are our estimates.

6.2. Real Hand Sequences

We have also tested our motion capturing algorithm on real hand motion sequences. Different schemes are compared for local motion capturing (Figure 5). The first one is a random search scheme in the $\mathcal{R}^7$ space with 5000 random samples. Since it makes use of no constraints, the performance is poor for local motion estimation and also degrades the global pose estimation. The second scheme uses CONDENSATION with 3000 samples in $\mathcal{R}^7$. It performs better than the first method, but it is still not robust enough. The third scheme is the proposed method, and it works accurately and robustly. The articulation model makes the computation more efficient and the local motion estimation enhances the accuracy of hand pose determination.

7. Conclusions

It is a difficult problem to capture both global hand poses and local finger articulations in video sequences because of the high degrees of freedom involved. This paper presents a divide and conquer approach to this problem by separating the global and local hand motion and estimating each component separately. For the global motion, we approximate the palm as a rigid planar object and use ICP to track the hand pose. The local finger articulation is tracked through a sequential Monte Carlo technique. The iterations between the estimates of global and local finger motion result in an accurate motion estimation.

Our current technique assumes clean backgrounds, which largely simplifies the image observation processes. We shall investigate the problem of clutter backgrounds. Also, a better hand model could be used to better handle out of plane rotations. Our current method requires manual initialization for tracking. It would be more interesting if we could achieve automatic initialization for the tracking.

Appendix: Solving $\mathbf{R}$ and $\mathbf{t_3}$ from $\mathbf{B}$

After we solve the motion based on the factorization method, $\mathbf{M}$ contains only $\mathbf{B}$ if we assume orthographic projection. In the following, we show how to estimate the rotation $\mathbf{R}$ and the depth translation t_3 from $\mathbf{B}$ under scaled orthographic projection. It is clear that

$$\mathbf{R} = \begin{bmatrix} t_3B_{11} & t_3B_{12} & R_{13} \\ t_3B_{21} & t_3B_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

From the property $\mathbf{R}^T\mathbf{R} = \mathbf{I}$, we obtain

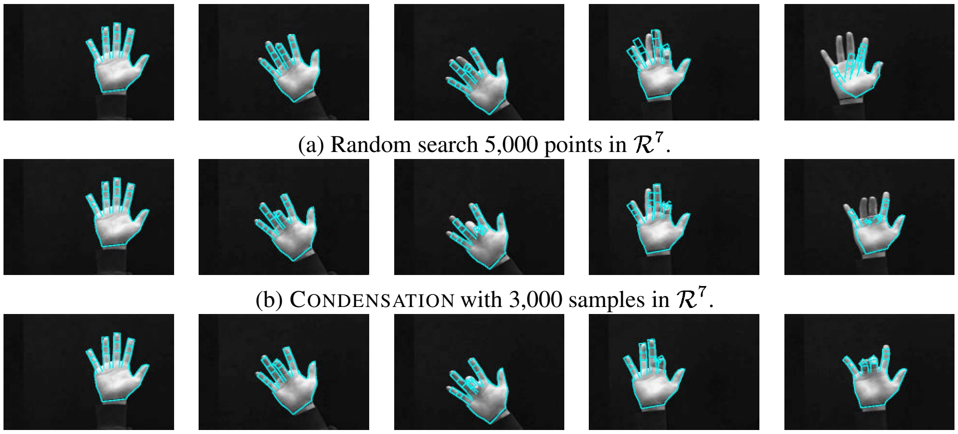
$$t_3^2b_1^2 + R_{31}^2 = 1 \quad \text{with } b_1^2 = B_{11}^2 + B_{21}^2 \qquad (9)$$

$$t_3^2b_2^2 + R_{32}^2 = 1 \quad \text{with } b_2^2 = B_{12}^2 + B_{22}^2 \qquad (10)$$

$$t_3^2d + R_{31}R_{32} = 0 \quad \text{with } d = B_{11}B_{12} + B_{21}B_{22} \qquad (11)$$

From (9), we have

$$t_3^2 = \frac{1}{b_1^2}(1 - R_{31}^2) \qquad (12)$$



(a) Random search 5,000 points in $\mathcal{R}^7$.

(b) CONDENSATION with 3,000 samples in $\mathcal{R}^7$.

(c) Our approach with 100 samples.

Figure 5. Comparison of different methods on real hand sequences. Our method is more accurate than the other two methods.

Then, from (11), we obtain

$$R_{32} = \frac{d}{b_1^2} R_{31} - \frac{d}{b_1^2} \frac{1}{R_{31}} \tag{13}$$

Substitute them into (10), and we get

$$(b_1^2 b_2^2 - d^2) R_{31}^4 + (2d^2 - b_1^2 b_2^2 + b_1^4) R_{31}^2 - d^2 = 0 \tag{14}$$

We may have at most four solutions, but usually only two because R_{31}^2 should be always positive. Even the two solutions can be possibly resolved by checking whether the resulting matrix is a rotation matrix or a reflection (the determinant is equal to -1).

8 Acknowledgements

The authors would like to thank the reviewers for their constructive comments. This work was supported in part by National Science Foundation Grant IIS-01-38965, and in part by the NCSA Alliance Program.

References

[1] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pages 126–133, Hilton Head Island, South Carolina, 2000.

[2] T. Heap and D. Hogg. Towards 3D hand tracking using a deformable model. In *Proc. of IEEE Int'l Conf. Automatic Face and Gesture Recognition*, pages 140–145, Killington, VT, 1996.

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

[4] J. J. Kuch and T. S. Huang. Vision-based hand modeling and tracking for virtual teleconferencing and telecollaboration. In *Proc. of IEEE Int'l Conf. on Computer Vision*, pages 666–671, Cambridge, MA, June 1995.

[5] J. Lee and T. Kunii. Model-based analysis of hand posture. *IEEE Computer Graphics and Applications*, 15:77–86, Sept. 1995.

[6] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *Proc. of IEEE Int'l Conf. Computer Vision*, pages 612–617, 1995.

[7] R. Rosales, S. Sclaroff, and V. Athitsos. 3D hand pose reconstruction using specialized mappings. In *Proc. IEEE Int'l Conf. on Computer Vision*, Vancouver, Canada, July 2001.

[8] J. Segen and S. Kumar. Shadow gesture: 3d hand pose estimation using a single camera. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 479–485, 1999.

[9] N. Shimada, Y. Shirai, Y. Kuno, and J. Miura. Hand gesture estimation and model refinement using monocular camera - ambiguity limitation by inequality constraints. In *Proc. of the 3rd Conf. on Face and Gesture Recognition*, pages 268–273, 1998.

[10] B. Stenger, P. R. S. Mendonça, and R. Cippola. Model-based 3d tracking of an articulated hand. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Hawaii, 2002.

[11] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography - a factorized method. *Int'l Journal of Computer Vision*, 9:137–154, 1992.

[12] Y. Wu and T. S. Huang. Capturing articulated human hand motion: A divide-and-conquer approach. In *Proc. of IEEE Int'l Conf. Computer Vision*, pages 606–611, 1999.

[13] Y. Wu and T. S. Huang. View-independent recognition of hand postures. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pages 88–94, 2000.

[14] Y. Wu, J. Lin, and T. S. Huang. Capturing natural hand articulation. In *Proc. of IEEE Int'l Conf. Computer Vision*, volume II, pages 426–432, 2001.

[15] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int'l Journal of Computer Vision*, 13:119–152, 1994.