# Vve are intecnupen, the world's leading publisher of Open Access books <br> Built by scientists, for scientists 

## 4,800

Open access books available

154
Countries delivered to

## 122,000

International authors and editors

Our authors are among the

## TOP 1\%

most cited scientists

135M
Downloads

WEB OF SCIENCE ${ }^{\top}$
Selection of our books indexed in the Book Citation Index in Web of Science ${ }^{T M}$ Core Collection (BKCI)

# Interested in publishing with us? Contact book.department@intechopen.com 



# Articulated Hand Tracking by ICA-based Hand Model and Multiple Cameras 

Makoto Kato, Gang Xu \& Yen-Wei Chen<br>Ritsumeikan University Japan

## 1. Introduction

Vision has the great potential to give the computers the ability of collecting information. In this chapter we study the tracking and capturing of 3-D free hand motions by computers. The hand has no markers and no special devices and no special conditions are required.
This chapter presents three techniques for the vision-based tracking. The first one is the ICAbased motion analysis. The second is articulated hand motion tracking by multiple cameras. The third is particle filtering with prediction.
A human hand has many joints and its high dimensionality makes it difficult to model hand motions. To make things easier, it is important to represent a hand motion in a low dimensional space. Principal component analysis (PCA) has been proposed to reduce the dimensionality. However, the PCA basis vectors only represent global features, which are not optimal for representing intrinsic features. This chapter proposes an efficient representation of hand motions by independent component analysis (ICA). The ICA basis vectors represent local features, each of which corresponds to the motion of a particular finger. This representation is more efficient in modeling hand motions for tracking and recognizing hand-finger gestures in an image sequence.
This chapter also proposes a new technique to simultaneously estimate the global hand pose and the finger articulation imaged by multiple cameras. Tracking a free hand motion against a cluttered background is a difficult task. The first reason is that hand fingers are selfoccluding and the second reason is the high dimensionality of the problem. In order to solve these difficulties, we propose using calibrated multiple cameras and at the same time improving search efficiency by predicted particle filtering.
$\underset{0}{ }$ The effectiveness of our methods is demonstrated by tracking free hand motions in real $\underset{\sim}{\tilde{\sim}}$ image sequences. The method is easily expanded for tracking human body motions in 3D.

## 2. Related work

Recently, recognition of hand gestures and hand motion tracking has become an important $\mathrm{O}_{\mathrm{O}}$ issue in the field of human-computer interaction. Many researchers tried or are trying to $\underset{1}{ }$ create a method by camera.
$\bar{\varnothing}$ The hand tracking methods by camera can be divided into two categories. One is $\bigcirc$ O appearance-based, and the other is model-based.

In the appearance-based methods, mapping between image features and hand pose is established. Hand pose estimation is formulated as an image database indexing problem, where the closest matches for an input hand image are retrieved from a large database of synthetic hand images. Stenger et al. proposed a new framework for Bayesian tracking based on the tree representation of the large database, which is effective for tracking 3D articulated motions in front of cluttered background (Stenger et al., 2003). The problem with the appearance-based method is that it requires a very large database.
In contrast, the model-based methods use an deformable hand model. The hand pose at the current frame is estimated from the current image input and previous pose. The problem of using a hand model is the high dimensionality. The high dimensionality causes an exponentially high computational cost. Particle filtering is one of the most successful object tracking algorithms (Isard \& Blake, 1998). However, to keep tracking correctness especially for rapid motions, it needs a large number of particles. Since it is infeasible to maintain dense sampling in high dimensional state spaces, two methods have been proposed to solve these problems. One is to reduce the state dimensionality and the other is to improve sampling and to make better prediction.
To reduce the dimensionality, Zhou et al. proposed an eigen dynamic analysis (EDA) method and constructed a dynamic Bayesian network based on EDA to analyze the generative sequence of natural hand motions (Zhou \& Huang, 2003). Wu et al. presented a method to capture natural hand motions by PCA and showed tracking results by particle filtering (Wu et al., 2001).
This chapter proposes a new model-based method. The previous researchers (Zhou \& Huang, 2003) (Wu et al., 2001) have reduced the dimensionality of hand pose space by PCA. Then they have learned basis motions in the PCA space. In contrast, we directly reduce the dimensionality of hand motion space by PCA. However, at our approach, it is impossible to use the PCA basis vectors for particle filtering, since the PCA basis vectors represent global features. To solve this problem, we propose to perform ICA to extract local features.
ICA (Hyvarinen et al., 2001) is a way of finding a linear non-orthogonal coordinate system in any multivariate data. The goal is to perform a linear transformation which makes the resulting variables as statistically independent from each other as possible. ICA has been successfully applied to many applications of image analysis and pattern recognition, such as face recognition (Bartlett et al., 2002), sign-language classification, color indexing, classification of multi-spectral images, and edge detection. In the proposed ICA-based hand motion representation, the ICA basis vectors of hand motions correspond to the motions of a particular finger and they are statistically independent. The representation is very efficient at particle filtering, since we can directly use these basis vectors. Furthermore, a linear combination of these ICA basis vectors can actually represent any hand motions.
To improve sampling efficiency, Rui et al. propose Unscented Particle Filter (UPF) (Rui \& Chen, 2001). The UPF uses the unscented Kalman filter to generate sophisticated proposal distributions that seamlessly integrate the current observation, thus greatly improving the tracking performance. This method needs to establish a system dynamics model. As for our 26-DOF problem, it is even hard to establish the system dynamics model. Deutscher et al. propose annealed particle filtering which is modified for searches in high dimensional state spaces (Deutscher et al., 2000). It uses a continuation principle, based on annealing, to introduce the influence of narrow peaks in the fitness function, gradually. It is shown to be capable of recovering full articulated body motion efficiently. However, the experiment is
done against a black background. Bray et al. propose smart particle filtering which combines the Stochastic Meta-Descent (SMD), based on gradient descent with particle filtering (Bray et al., 2004). Their 3D hand tracking result is robust and accurate. However, they need depth maps generated by a structured light 3D sensor, which are not available in real time.
We propose to add prediction to particle filtering. Parameters in the next frame are predicted and more particles are accordingly generated for areas of higher likelihood. The method is straightforward but proven to very effective in significantly reducing search cost. Another problem with the model-based approach is self-occlusion. While a hand moves freely, parts of the hand change from being visible to being invisible, and then becoming visible again. Previously proposed techniques avoid this problem by restricting hand motions to only those that are frontal to the camera (Wu et al., 2001). To overcome this restriction, we propose to use multiple pre-calibrated cameras, so that parts invisible in one camera are still visible in at least another camera. While this is the right approach to the selfocclusion problem, more observations put more burdens on the already busy computer. This motivates further improvement of search efficiency. Although Rehg and Kanade (Rehg \& Kanade, 1995) proposed to use multiple cameras for finger tracking, hand motion in this chapter is much more complicated and we need to design and implement more efficient method.

## 3. Representation of hand motions

### 3.1 Hand model



Figure 1. (a): Hand model with the name of each joint, and the degrees of freedom (DOF) for each joint. (b) Hand model rendered in OpenGL.

In our study a human hand is rendered in OpenGL using spheres, cylinders, and rectangular parallelepiped. A human hand can be described in this way: the base is a palm and five fingers are attached to the palm. Each finger has four degrees of freedom (DOF). Two of four DOF correspond to the metacarpophalangeal joint (MP) and its abduction (ABD). The other two correspond to the proximal interphalangeal joint (PIP) and the distal interphalangeal joint (DIP) (Lee \& Kunii, 1995). It is shown in Fig. 1. Therefore, our hand model has 20 DOF. In addition, to represent the position and orientation of a hand, we need 6 more parameters, 3 for position and 3 for orientation. In total, the hand model has 26 DOF.

### 3.2 Hand motion data

The data of hand motions is captured by a data glove. It is collected starting from the openpalmed, 5 -fingers extended position, with the fingers moving to various combinations of touching the palm. Since a hand consists of five fingers, 31 different hand motions are captured as:

$$
\begin{equation*}
{ }_{5} C_{5}+{ }_{5} C_{4}+{ }_{5} C_{3}+{ }_{5} C_{2}+{ }_{5} C_{1}=31 \tag{1}
\end{equation*}
$$

where $C$ means combination.
Angles of 20 DOF of 15 joints are measured. We divide the motion data of each DOF into 100 instants along the time axis. Then the motion of each DOF can be represented by a row vector of 100 -dimensions. We arrange the thumb, index, middle, ring, and pinkie in order, where each finger consists of four DOF which are MP, PIP, DIP, and ABD in order as Fig. 2. We define this 2000-dimensional row vector as a hand motion vector $\mathbf{x}_{i}, \quad i=1, \cdots, 31$.


Figure 2. Example of a hand motion vector $\mathbf{x}_{i}$. This vector represents the hand motion, where an index finger is bended intentionally. The numbers on $x$-axis refer to time in each DOF. The numbers on $y$-axis refer to angles.

### 3.3 Constraints of hand motion

Analysis of hand motion is a task of high cost, because the joint angle space of hand is $\Theta \subset \mathfrak{R}^{20}$. Fortunately, a hand motion has certain constraints. One type of constraints is the so-called static constraints in literature (Lee \& Kunii, 1995), which define limits on the ranges of finger motions such as $0^{\circ} \leq \theta_{M P} \leq 90^{\circ}$. These constraints limit hand motions within a boundary in $\mathfrak{R}^{20}$. However, these constraints can not be used to reduce the dimensionality.
Another type of constraints describes the correlations among different joints, and we can use this type of constraints to reduce the dimensionality of hand motions. For example, the
motions of the DIP joint and PIP joint are generally not independent and they can be described as $\theta_{D I P}=2 / 3 \theta_{P I P}$ from the study of biomechanics.

### 3.4 Dimensionality reduction by PCA

The purpose of PCA is to find a smaller set of variables with less redundancy. The redundancy is measured by correlations between data elements, i.e.

$$
\begin{equation*}
\mathbf{r}_{i}=\mathbf{P}\left(\mathbf{x}_{i}-\mathbf{x}_{0}\right)^{T} \tag{2}
\end{equation*}
$$

where $\mathbf{P}$ is the transformation matrix. Each row of $\mathbf{P}$ corresponds to the first several basis vectors, which are calculated from the sample data set using Singular Value Decomposition (SVD). $\mathbf{x}_{0}=1 / 31 \sum_{k=1}^{31} \mathbf{x}_{k}$ is the mean of the data set.
Fig. 3 is the rendered hand motions along the PCA basis vectors. We can see that a hand motion along the PCA basis vector represents a global finger motion which means that fingers move together. Most of hand motions along the PCA basis vectors are unfeasible hand motions.


Figure 3. Rendered hand motions along the PCA basis vectors from frontal view (view1) and tilted view (view2). (a)-(e) corresponds to the first five PCA basis vectors respectively.

### 3.5 Representation of hand motion by ICA

Although PCA is efficient for dimensionality reduction, it has difficulty representing the intrinsic features, because its basis vectors represent global features. In order to solve this problem, we use ICA to represent hand motions. First, we perform PCA to reduce the dimensionality. Then we perform ICA to extract intrinsic features. ICA is a generalized technique of PCA and has proven to be an effective tool of feature extraction.


Figure 4. Rendered hand motions along the ICA basis vectors from frontal view (view1) and tilted view (view2). (a)-(e) corresponds to the five ICA basis vectors respectively.
A hand motion vector $\mathbf{X}_{i}$ can be represented by a linear combination of basis vectors as

$$
\begin{equation*}
\mathbf{x}_{i}=\sum_{j=1}^{N} a_{i j} \mathbf{u}_{j}=a_{i 1} \mathbf{u}_{1}+a_{i 2} \mathbf{u}_{2}+\cdots+a_{i N} \mathbf{u}_{N} \tag{3}
\end{equation*}
$$

where $\mathbf{u}_{j}$ is the $j$-th independent basis vector and $a_{i j}$ is the $j$-th coefficient. Equation (3) can then be written as follows in matrix form:

$$
\begin{equation*}
\mathbf{X}=\mathbf{A} \mathbf{U} \tag{4}
\end{equation*}
$$

where $\mathbf{A}$ is the mixing matrix, producing a matrix $\mathbf{X}$ with hand motion vectors in its row. Because we want to obtain $\mathbf{U}$ from sample hand motions $\mathbf{X}$ alone, the problem is actually the Blind Source Separation (BSS) problem, which can be solved by ICA as

$$
\begin{equation*}
\hat{\mathbf{U}}=\mathbf{W X} \tag{5}
\end{equation*}
$$

The goal of ICA is to find the unmixing matrix $\mathbf{W}$ such that the rows of $\hat{\mathbf{U}}$ are as statistically independent as possible. Several ICA algorithms have been proposed. Here we use the infomax algorithm proposed by Bell and Sejnowski (Bell \& Sejnowski, 1995), which was successfully used in face recognition (Bartlett et al., 2002). The approach is to maximize the joint entropy by using stochastic gradient ascent. The gradient update rule for the weight matrix $\mathbf{W}$ is as follow:

$$
\begin{equation*}
\Delta \mathbf{W}=\left(\mathbf{I}+g(\mathbf{U}) \mathbf{U}^{T}\right) \mathbf{W} \tag{6}
\end{equation*}
$$

where $\mathbf{U}=\mathbf{W X}$ and $g(u)=1-2 /\left(1+e^{-u}\right)$.
Fig. 4 is the rendered hand motions along the ICA basis vectors. Compared with PCA basis vectors as Fig. 3, a hand motion along the ICA basis vector represents a local finger motion which corresponds to a particular finger motion. We can see that only one finger moves dominantly, while other fingers move very little. Furthermore, hand motions along the ICA basis vectors are feasible hand motions.

### 3.6 Efficient representation of hand pose

The ICA-based model can represent a hand pose by five independent parameters, each of which corresponds to a particular finger motion at a particular time instant respectively. This is shown in Fig. 5. It can also expressed formally as follows:

$$
\begin{align*}
\operatorname{HandPose}= & \text { ThumbMotion }\left(t_{1}\right)+\operatorname{IndexMotion}\left(t_{2}\right)+\operatorname{MiddleMotion}\left(t_{3}\right) \\
& +\operatorname{RingMotion}\left(t_{4}\right)+\operatorname{PinkieMotion}\left(t_{5}\right) \\
= & I \operatorname{CAbasis} 5\left(t_{1}\right)+I \operatorname{CAbasis} 3\left(t_{2}\right)+I \operatorname{CAbasis} 1\left(t_{3}\right) \\
& +I \operatorname{CAbasis} 4\left(t_{4}\right)+I \operatorname{CAbasis} 2\left(t_{5}\right) \tag{7}
\end{align*}
$$

Thus, any hand gesture can then be represented by 5 parameters $t_{1}, t_{2}, t_{3}, t_{4}, t_{5}$.



Figure 5. Each row represents the hand motion along the ICA basis vector. A hand pose is determined by five parameters t1-t5 which refer to time. The ICA basis 1 corresponds to a middle finger motion, the ICA basis 2 corresponds to a pinkie motion, the ICA basis 3
corresponds to an index finger motion, the ICA basis 4 corresponds to a ring finger motion, and the ICA basis 5 corresponds to a thumb motion.

### 3.7 Justification of the ICA-based hand model

When we solve the blind source separation problem, we need to know the number of source signals. To verify the 5-dimension ICA basis is sufficient to represent the finger pose and motion, we use "leave-one-out cross-validation". The result is shown in Fig. 6. In the figure, the vertical axis indicates root mean square (RMS) error, and the horizontal axis indicates the number of the ICA basis vectors used to recover a hand motion data. The average of RMS error is plotted along the vertical axis. The error bars shows maximum error and minimum error.


Figure 6. Leave-one-out cross-validation on the hand motion data set.


(d) ICA-basis 4

(e) ICA-bafic 5

Figure 7. Five ICA basis vectors.
The five ICA basis vectors are shown in Fig. 7. ICA can be applied only if the independent components are "statistically independent" and also obey "non-gaussian" distribution. Here, the independent components are the ICA basis vectors. We calculate the covariance matrix of the independent components to verify statistically independent. The covariance matrix is
$\square\left[\begin{array}{ccccc}7.6711 & -0.1993 & -0.3399 & 0.1826 & 0.162 \\ -0.1933 & 6.5648 & -0.1007 & 0.3804 & 0.3889 \\ -0.3399 & -0.1007 & 6.2993 & 0.1464 & 0.0071 \\ 0.1826 & 0.3804 & 0.1464 & 6.1052 & -0.3711 \\ 0.162 & 0.3889 & 0.0071 & -0.3711 & 4.4959\end{array}\right]$

The covariance matrix is almost diagonal, which implies the independent components are statistically independent.
In order to measure "non-gaussianity" of the resulting independent components, we calculate the kurtosis of each independent component. Since the kurtosis of Gaussian distribution is equal to 0 , we can measure "non-gaussianity" by calculating the kurtosis. The normalized kurtosis (Hyvarinen et al., 2001) is defined as

$$
\begin{equation*}
\frac{E\left\{x^{4}\right\}}{\left[E\left\{x^{2}\right\}\right]^{2}}-3 \tag{9}
\end{equation*}
$$

The normalized kurtosis of the independent components is shown in Table 1. Note that the normalized kurtosis of Gaussian distribution is equal to 0 , that of Laplace distribution is 3 , and that of exponential distribution is 6 .

|  | Normalized kurtosis |
| :---: | :---: |
| ICA basis 1 | 16.6535 |
| ICA basis 2 | 9.7997 |
| ICA basis 3 | 13.1544 |
| ICA basis 4 | 9.0106 |
| ICA basis 5 | 4.3983 |

Table 1. Normalized kurtosis of each component.

## 4. Hand tracking by particle filtering

### 4.1 Particle filtering

The particle filtering algorithm (Djuric et al., 2003) is a sequential Monte Carlo method. The algorithm is powerful in approximating non-Gaussian probability distributions. Particle filtering is based on sequential importance sampling and Bayesian theory. With particle filtering, continuous distributions are approximated by discrete random sample sets, which are composed of weighted particles. The particles represent hypotheses of possible solutions and the weights represent likelihood.
There are three main steps in the algorithm: resampling, diffusion, and observation. The first step selects the particles for reproduction. In this step, particles that have heavier weights are more likely to be selected. Heavy-weight particles generate new ones, while light-weight particles are eliminated. The second step diffuses particles randomly. A part of space that is more likely to have a solution has more particles, while a part of space that is less likely to have a solution has fewer particles. The third step measures the weight of each particle according to an observation density. Fig. 8 shows a pictorial description of particle filtering.


Figure 8. One time-step in particle filtering. There are 3 steps, resampling-diffusionobservation.

### 4.2 Generating particles

We implement particle filtering for tracking articulated hand motions. According to the Bayes rule, the hand pose of the current frame $\mathbf{x}_{t}$ can be estimated from the prior hand pose $\mathbf{x}_{t-1}$ as

$$
\begin{equation*}
p\left(\mathbf{x}_{t} \mid z_{t}\right) \propto p\left(z_{t} \mid \mathbf{x}_{t}\right) p\left(\mathbf{x}_{t} \mid z_{t-1}\right) \tag{10}
\end{equation*}
$$

where $z_{t}$ is the observation of the current frame.
The important part of particle filtering is to generate particles. In order to represent a posteriori $p\left(\mathbf{x}_{t} \mid z_{t}\right)$, we employ a time-stamped sample set, denoted $\left\{\mathbf{s}_{t}^{(n)}, n=1, \cdots, N\right\}$, which is weighted by the observation density $\pi_{t}^{(n)}=p\left(z_{t} \mid \mathbf{x}_{t}=\mathbf{s}_{t}^{(n)}\right)$. The weights $\pi_{t}^{(n)}$ are normalized so that $\sum_{N} \pi_{t}^{(n)}=1$. Then the sample set $\left\{\mathbf{s}_{t}^{(n)}, \pi_{t}^{(n)}\right\}$ represents the posteriori $p\left(\mathbf{x}_{t} \mid z_{t}\right)$. The sample set is propagated from $\left\{\mathbf{s}_{t-1}^{(n)}, \pi_{t-1}^{(n)}\right\}$ which represents $p\left(\mathbf{x}_{t-1} \mid z_{t-1}\right)$. A prior $p\left(\mathbf{x}_{t} \mid z_{t-1}\right)$ can be represented as

$$
\begin{equation*}
p\left(\mathbf{x}_{t} \mid z_{t-1}\right)=\int_{\mathbf{x}_{t-1}} p\left(\mathbf{x}_{t} \mid \mathbf{x}_{t-1}\right) p\left(\mathbf{x}_{t-1} \mid z_{t-1}\right) \tag{11}
\end{equation*}
$$

Then random samples are drawn along each ICA basis vector, i.e.,

$$
\begin{equation*}
s_{t}^{(n)} \sim p\left(s_{t} \mid s_{t-1}\right)=N\left(s_{t-1} \mid \sigma\right) \tag{12}
\end{equation*}
$$

For finger motions, we can make samples along each ICA basis vector shown in Fig. 12 due to the efficient representation of hand pose by the ICA-based model. A finger motion is determined by five parameters in the ICA-based model which has five dimensions. Other parameters are position and rotation. They are presented by translation $t_{x}, t_{y}, t_{z}$ and


Figure 9. Generating hypotheses along each ICA basis vector. Black points indicate current sample $S_{t}$, while white circles indicate hypotheses $S_{t+1}$.
rotation $r_{x}, r_{y}, r_{z}$. They also propagate as

$$
\begin{gather*}
s_{t}^{(n)} \sim p\left(s_{t} \mid s_{t-1}\right)=N\left(s_{t-1} \mid \sigma_{\text {translation }}\right)  \tag{13}\\
s_{t}^{(n)} \sim p\left(s_{t} \mid s_{t-1}\right)=N\left(s_{t-1} \mid \sigma_{\text {rotation }}\right) \tag{14}
\end{gather*}
$$

Then the total dimensionality of $\mathbf{s}_{t}$ is 11, including 5 for finger motion, 3 for translation, and 3 for rotation.

## 5. Occlusion-free tracking by multiple cameras

### 5.1 Tracking by multiple cameras

We perform camera calibration so that the intrinsic parameters and positions and orientations of the cameras recovered (Zhang, 1999). Once the cameras are calibrated, a hand model is projected onto the images, and the projected images are compared with real observations so that the parameters of the hand model can be estimated. Since calibrated cameras do not increase unknown parameters, more images do not mean more parameters. They merely bring more information.

In our currently experiments, we use two cameras looking at the hand, with the two cameras separated by roughly 90 degrees. This brings a great improvement over using a single camera, and is sufficient in handling occlusions.

### 5.2 Relation between the hand model and two cameras

The relation between two cameras is drawn as follows. The 3D coordinate system centered at optical center of camera 1 is $\mathbf{X}$. The 3D coordinate system centered at optical center of


Figure 10. Relation between two cameras.
camera 2 is $\mathbf{X}^{\prime}$. As depicted in Fig. 10, the rotation matrix and the translation vector from the coordinate system of camera 1 to the coordinate system of camera 2 are $\mathbf{R}_{c}, \mathbf{t}_{c}$. Then the relation between the two coordinate systems is given by

$$
\begin{equation*}
\mathbf{X}=\mathbf{R}_{c} \mathbf{X}^{\prime}+\mathbf{t}_{c} \tag{15}
\end{equation*}
$$

The 3D coordinate system of hand model is $\mathbf{X}_{m}$. The rotation matrix and the translation vector from the coordinate system of hand model to the coordinate system of camera 1 are $\mathbf{R}_{w}, \mathbf{t}_{w}$. Then the relation between the two coordinate systems is given by

$$
\begin{equation*}
\mathbf{X}_{m}=\mathbf{R}_{w} \mathbf{X}+\mathbf{t}_{w} \tag{16}
\end{equation*}
$$

From (15) and (16), we can transform $\mathbf{X}_{m}$ to $\mathbf{X}$ and $\mathbf{X}^{\prime}$, and then project the hand model onto the images.

### 5.3 Observation model

We employ edge and silhouette information to evaluate the hypotheses. For edge information, we employ the Chamfer distance function (Stenger et al., 2003). First, we perform Canny edge detection to the input image. In the result image of edge detection, the edge pixels are black and other pixels are white. Then, at each pixel, we calculate the distance from each pixel to the closest edge point by using distance transformation. If the distance is over a threshold, the distance is set to the threshold. A distance map of the input image is obtained. Fig. 11 (b) shows an example of distance map. Then we project the edge of the hand model onto the distance map. We add all distances along the edge points of the projected hand model, and calculate the average of distances. Then the likelihood from the edge information is


Figure 11. (a) Input image (b) Distance map of edge observation (c) Extracted silhouette

(c)

Figure 12. Areas of silhouette measurements. Black areas are the corresponding areas. (a) $a_{I}-a_{O}$, (b) $a_{M}-a_{O}$, (c) $a_{I}-a_{M}$. Note that $a_{I}$ is the silhouette of input image, $a_{M}$ is the silhouette of hand model, and $a_{O}$ is the silhouette of overlap.

$$
\begin{equation*}
p_{\text {edge }}\left(z_{t} \mid \mathbf{x}_{t}\right) \propto \exp \left[-\frac{(\text { averageDist })^{2}}{2 \sigma_{\text {edge }}^{2}}\right] \tag{17}
\end{equation*}
$$

where averageDist is the average of distances.
In order to extract the silhouette of a hand region, we convert image color space from RGB to HSV (hue, saturation and brightness). Then the skin color region is extracted by using a threshold. Fig. 11 (c) shows an extracted silhouette. We calculate subtractions of the area of silhouette. The three calculated subtraction results are shown in Fig. 12. The subtractions of $a_{I}-a_{O}$ and $a_{M}-a_{O}$ are used to measure the similarity of the hand position. The subtraction of $a_{I}-a_{M}$ is used to measure the similarity of hand finger pose. Then likelihoods from the silhouette information are

$$
\begin{align*}
& p_{\text {sil_IO }}\left(z_{t} \mid \mathbf{x}_{t}\right) \propto \exp \left[-\frac{\left(a_{I}-a_{O}\right)^{2}}{2 \sigma_{s i l_{-} I O}^{2}}\right]  \tag{18}\\
& p_{\text {sil_MO }}\left(z_{t} \mid \mathbf{x}_{t}\right) \propto \exp \left[-\frac{\left(a_{M}-a_{O}\right)^{2}}{2 \sigma_{\text {sil_MO }}^{2}}\right]  \tag{19}\\
& p_{\text {sil_IM }}\left(z_{t} \mid \mathbf{x}_{t}\right) \propto \exp \left[-\frac{\left(a_{I}-a_{M}\right)^{2}}{2 \sigma_{\text {sil_IM }}^{2}}\right] \tag{20}
\end{align*}
$$

Thus the final likelihood is

When we use multiple cameras, the likelihood is

$$
\begin{equation*}
p\left(z_{t} \mid \mathbf{x}_{t}\right) \propto \prod_{i=1}^{n} p_{i}\left(z_{t} \mid \mathbf{x}_{t}\right) \tag{22}
\end{equation*}
$$

where $n$ is the number of cameras.

### 5.4 Particle filtering with prediction

The classical particle filtering requires an impractically large number of particles to follow rapid motions and to keep tracking correct. It becomes a serious problem when the tracking target has a high dimensional state space like hand tracking. In order to tackle this problem, we propose using prediction to generate better proposal distributions.

According to the Bayes rule, the hand pose of the current frame $\mathbf{X}_{t}$ can be estimated from the prior hand pose $\mathbf{x}_{t-1}$ as

$$
\begin{equation*}
p\left(\mathbf{x}_{t} \mid z_{1: t}\right) \propto p\left(z_{t} \mid \mathbf{x}_{t}\right) p\left(\mathbf{x}_{t} \mid z_{1: t-1}\right) \tag{23}
\end{equation*}
$$

where

$$
\begin{equation*}
p\left(\mathbf{x}_{t} \mid z_{1: t-1}\right)=\int_{\mathbf{x}_{t-1}} p\left(\mathbf{x}_{t} \mid \mathbf{x}_{t-1}\right) p\left(\mathbf{x}_{t-1} \mid z_{1: t-1}\right) \tag{24}
\end{equation*}
$$

$z_{t}$ is the observation of the current frame, $p\left(z_{t} \mid \mathbf{x}_{t}\right)$ is the likelihood distribution and $p\left(\mathbf{x}_{t} \mid \mathbf{x}_{t-1}\right)$ is the transition probability distribution. (23) can be interpreted as the equivalent of the Bayes rule:

$$
\begin{equation*}
p(\mathbf{x} \mid z) \propto p(z \mid \mathbf{x}) p(\mathbf{x}) \tag{25}
\end{equation*}
$$



Figure 13. Five-finger tracking with 50 particles by our method. The projection of OpenGL hand model's edge is drawn on the images


Figure 14. Examples of the corresponding OpenGL hand model of Fig. 13.

In particle filtering, the sequence of probability distributions is approximated by a large set of particles. Therefore, how to propagate the particles efficiently in areas of higher likelihood significantly affects tracking results. The particles are defined as follows: in order to represent a posteriori $p\left(\mathbf{x}_{t} \mid z_{1: t}\right)$, we employ a time-stamped sample set, denoted $\left\{\mathbf{s}_{t}^{(n)}, n=1, \cdots, N\right\}$. The sample set is weighted by the observation density $\pi_{t}^{(n)}=p\left(z_{t} \mid \mathbf{x}_{t}=\mathbf{s}_{t}^{(n)}\right)$, where the weights $\pi_{t}^{(n)}$ are normalized so that $\sum_{N} \pi_{t}^{(n)}=1$. Then the sample set $\left\{\mathbf{s}_{t}^{(n)}, \pi_{t}^{(n)}\right\}$ represents the posteriori $p\left(\mathbf{x}_{t} \mid z_{1: t}\right)$. The sample set of the posteriori is propagated from $\left\{\mathbf{s}_{t-1}^{(n)}, \pi_{t-1}^{(n)}\right\}$ which represents $p\left(\mathbf{x}_{t-1} \mid z_{1: t-1}\right)$ as shown in Fig. 6. The transition probability distribution $p\left(\mathbf{x}_{t} \mid \mathbf{x}_{t-1}\right)$ affects $p\left(\mathbf{x}_{t} \mid z_{1: t-1}\right)$, which in turn affects $p\left(\mathbf{x}_{t} \mid z_{1: t}\right)$.
$p\left(\mathbf{x}_{t} \mid \mathbf{x}_{t-1}\right)$ is modeled by a dynamical model. The simplest dynamical model is

$$
\begin{equation*}
\mathbf{s}_{t}^{(n)}=\mathbf{s}_{t-1}^{(n)}+\mathbf{B} \tag{26}
\end{equation*}
$$

where $\mathbf{B}$ is a multivariate Gaussian distribution with covariance $\mathbf{P}$ and mean $\mathbf{0}$. However, this simple dynamical model does not propagate the particles efficiently and many particles are wasted in areas of lower likelihood.
To overcome these difficulties, we simply use the first-order approximation of Taylor series expansion for prediction:


Figure 15. Demonstration of finger tracking with 200 particles. The projection of OpenGL hand model's edge is drawn on images.


Figure 16. Examples of the corresponding OpenGL hand model of Fig. 15.

$$
\begin{equation*}
\mathbf{s}_{t}^{(n)}=\mathbf{s}_{t-1}^{(n)}+\frac{\partial \mathbf{s}_{t-1}^{(n)}}{\partial t} \Delta t+\mathbf{B} \tag{27}
\end{equation*}
$$

We also tried to use the second-order approximation of Taylor series expansion

$$
\begin{equation*}
\mathbf{s}_{t}^{(n)}=\mathbf{s}_{t-1}^{(n)}+\frac{\partial \mathbf{s}_{t-1}^{(n)}}{\partial t} \Delta t+\frac{1}{2} \frac{\partial^{2} \mathbf{s}_{t-1}^{(n)}}{\partial t^{2}} \Delta t^{2}+\mathbf{B} \tag{28}
\end{equation*}
$$

However, the tracking gets trapped in local minima. The reason is that the second derivative cannot be estimated accurately due to noise.

## 6. Experimental results

The proposed algorithm has been tested on real image sequences. We collect training data using a data glove. The training data is 31 different hand motions as described in Chapter 4. At first, PCA is applied to reduce the dimensionality. Then ICA is applied to obtain the ICA basis vectors. We applied our tracking algorithm on real image sequences. In the experiment, we assume that the hand model is roughly matched with the hand at the first frame. Then our tracking algorithm automatically track hand motions. The hand model is manually initialized to fit finger length and palm size. The experimental results demonstrate the effectiveness of our method by tracking hand in real image sequences. The video sequence is available from http://www.cvg.is.ritsumei.ac.jp/ $\sim$ kmakoto/.


Figure 17. Two image sequences (a) and (b). (a) is taken by the left camera. (b) is taken by the right camera. The image sequences include rapid motion, large rotations angle against a camera, occlusions and a cluttered background

### 6.1 Tracking local motions by one camera

We did two experiments by using one camera. In the first experiment, we use 50 particles per frame. Fig. 13 shows some frames of video sequence. Fig. 14 shows some corresponding hand models. The second experiment includes some local finger motions including rock-paper-scissors. We use 200 particles per frame. Fig. 15 shows some frames of the video sequence and Fig. 16 shows some corresponding OpenGL hand models.
Experimental results show that the ICA-based model is very useful for articulated hand tracking in image sequences since all hand motions can be represented by only 5 parameters and each parameter corresponds to a particular finger motion in the ICA-based model.

### 6.2 Occlusion-free tracking by multiple cameras

The tracking by one camera has some limitations. One of the limitations is caused by occlusions. The other limitation is caused by extreme changes in rotation angle toward a camera. One solution to the problem is tracking by multiple cameras. The following two image sequences (Fig. 17) include rapid motion, large rotations angle against a camera, occlusions and a cluttered background.


Frame 50


Frame 60
(a)


Frame 50


Frame 60
(b)

Figure 18. (a) Tracking result by the right camera only. (b) Tracking result by the left camera only.

At first, we tried the experiment by a single camera to two image sequences respectively, shown in Fig. 18 (a) and Fig. 18 (b). In Fig. 18 (a), at frame 50, the hand orientation is slightly incorrect and then the error becomes larger, finally, at frame 60 , the hand orientation is completely incorrect. In Fig. 18 (b), at frame 50, the hand orientation is incorrect and then the error becomes larger, finally at frame 60, the tracking estimated that the hand fingers exist at the hand wrist position.
Fig. 19 shows the tracking result by multiple cameras. The experiment was run using 10000 particles per frame. The tracking correctly estimated hand position and motion throughout the sequence.
From the results, we can see that occlusion is a severe problem for tracking by a single camera but is not a problem for multiple cameras.

(a)

(b)

Figure 19. Tracking result by two cameras. (a) Camera 1 view. (b) Camera 2 view.

The projection of hand model's edge is drawn on the images by red lines. The CG models are examples of some corresponding hand models

### 6.3 Tracking with prediction

In this experiment, we compared the methods with prediction and without prediction. Fig. 20 (a) is the result without prediction and Fig. 20 (b) is that with prediction. In Fig 20 (a), at frame 50, the hand orientation is slightly incorrect, and then the error becomes larger and finally, at frame 60, the tracking estimated that the hand is upside down comparing with the real hand. While with the prediction, we can avoid such kind of error (Fig. 20 (b)).


Figure 20. (a) Tracking result without prediction. (b) Tracking result with prediction.


- Without Prediction:
- Without Prediction:
10000 particles
10000 particles
-With Prediction: }50
-With Prediction: }50
particles
particles
With Prediction: 3000
With Prediction: 3000
particles
particles
With Prediction: }1000
With Prediction: }1000
particles
particles
-With Prediction: }1500
-With Prediction: }1500
particles
particles

Figure 21. Trajectory of the rotation around $Y$ axis (unit: degrees).

### 6.4 The number of particles

We also did experiments with different numbers of particles per frame in order to find out how many particles are suitable for the tracking. We show the trajectory of the rotation around Y axis in Fig. 21.
We did the experiment with 500 particles, 3000 particles, 10000 particles and 15000 particles. The results have dramatic change when we increase the number of particles from 500 to 10000. And the results only have slight change when we increase the number of particles
from 10000 to 15000 . Therefore, 10000 is the optimized number of particles for this hand motion.

## 7. Conclusion

In this chapter, we proposed three new approaches, the ICA-based hand model, articulated hand motion tracking by multiple cameras, and Particle filtering with prediction.
The ICA-based hand model is the ICA-based representation of hand articulation for tracking hand-finger gestures in image sequences. The dimensionality of the hand motion space is reduced by PCA and then ICA is applied to extract the local feature vectors. In the ICAbased model, each of the first five basis vectors corresponds to a particular finger motion, because the joints in each finger have stronger dependencies than the joints across different fingers. In the ICA-based model, hand poses can be represented by five parameters with each parameter corresponding to a particular finger motion. We implemented articulated hand motion tracking by particle filter using this ICA-based hand model. Experimental results show that the ICA-based model is very useful for articulated hand tracking in image sequences.
Next approach is an articulated hand motion tracking by multiple cameras. This method is useful for gesture recognition. Tracking a free hand motion against a cluttered background was unachievable in previous methods because hand fingers are self-occluding. To improve search efficiency, we proposed adding prediction to particle filtering so that more particles are generated in areas of higher likelihood. The experimental results show that our method can correctly and efficiently track the hand motion throughout the image sequences even if hand motion has large rotation against a camera.
The methods in this chapter are easily extended to many other visual motion capturing tasks.

## 8. References

Bartlett, M. S.; Movellan, J. R. \& Sejnowski, T. J. (2002). Face Recognition by Independent Component Analysis, IEEE Trans. Neural Networks, vol.13, no.6, pp.1450-1464, Nov. 2002.

Bell, A. J. \& Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution, Neural Computation, vol.7, no.6, pp.1129-1159, 1995.

Bray, M.; Meier, E. K. \& Gool, L. V. (2004). Smart Particle Filtering for 3D Hand Tracking, Proceedings of Int'l Conf. Face and Gesture recognition, pp. 675-680, 2004.
Deutscher, J.; Blake, A. \& Reid, I. (2000). Articulated Body Motion Capture by Annealed Particle Filtering, Proceedings of Computer Vision and Pattern Recognition, Vol. 2, pp. 126-133, 2000.
Djuric, P. M.; Kotecha, J. H.; Zhang, J.; Huang, Y.; Ghirmai, T.; Bugallo, M. F. \& Miguez, J. (2003). Particle filtering, IEEE Signal Processing Magazine, pp.19-38, Sept. 2003.

Hyvarinen, A.; Karhunen, J. \& Oja, E. (2001). Independent Component Analysis. Wiley, 9780471405405, New York.
Isard, M. \& Blake, A. (1998). CONDENSATION-conditional density propagation for visual tracking, Int. J.Computer Vision, 1998.

Lee, J. \& Kunii, T. (1995). Model-based analysis of hand posture, IEEE Computer Graphics and Applications, 15: pp.77-86, Sept. 1995.
Rehg, J. \& Kanade, T. (1995). Model-Based Tracking of Self Occluding Articulated Objects, Proceedings of Int'l Conf. Computer Vision, pp. 612-617, 1995.
Rui, Y. \& Chen, Y. (2001). Better Proposal Distributions: Object Tracking Using Unscented Particle Filter, Proceedings of Computer Vision and Pattern Recognition, Vol. 2, pp. 786793, 2001.
Sminchisescu, C. \& Triggs, B. (2001). Covariance Scaled Sampling for Monocular 3D Body Tracking, Proceedings of Computer Vision and Pattern Recognition, Vol. 1, pp. 447-454, 2001.

Stenger, B.; Thayananthan, A.; Torr, P. H. S. \& Cipolla, R. (2003). Filtering using a tree-based estimator, Proceedings of Int'l Conf. Computer Vision, pp.1102-1109, Nice, France, October 2003.
Wu, Y.; Lin, J. \& Huang, T. S. (2001). Capturing natural hand articulation, Proceedings of Int'l Conf. Computer Vision, pp.426-432, Vancouver, July 2001.
Zhang, Z. (1999). Flexible Camera Calibration by Viewing a Plane from Unknown Orientations, Proceedings of Int'l Conf. Computer Vision, pp.666-673, Greece, Sept 1999.

Zhou, H. \& Huang, T. S. (2003). Tracking Articulated Hand Motion with Eigen Dynamics Analysis, Proceedings of Int'l Conf. Computer Vision, pp.1102-1109, Nice, France, October 2003.


# Scene Reconstruction Pose Estimation and Tracking <br> Edited by Rustam Stolkin 

ISBN 978－3－902613－06－6
Hard cover， 530 pages
Publisher I－Tech Education and Publishing
Published online 01，June， 2007
Published in print edition June， 2007

This book reports recent advances in the use of pattern recognition techniques for computer and robot vision． The sciences of pattern recognition and computational vision have been inextricably intertwined since their early days，some four decades ago with the emergence of fast digital computing．All computer vision techniques could be regarded as a form of pattern recognition，in the broadest sense of the term．Conversely if one looks through the contents of a typical international pattern recognition conference proceedings，it appears that the large majority（perhaps $70-80 \%$ ）of all pattern recognition papers are concerned with the analysis of images．In particular，these sciences overlap in areas of low level vision such as segmentation， edge detection and other kinds of feature extraction and region identification，which are the focus of this book．

## How to reference

In order to correctly reference this scholarly work，feel free to copy and paste the following：

Makoto Kato，Gang Xu and Yen－Wei Chen（2007）．Articulated Hand Tracking by ICA－based Hand Model and Multiple Cameras，Scene Reconstruction Pose Estimation and Tracking，Rustam Stolkin（Ed．），ISBN：978－3－ 902613－06－6，InTech，Available from：
http：／／www．intechopen．com／books／scene＿reconstruction＿pose＿estimation＿and＿tracking／articulated＿hand＿trac king＿by＿ica－based＿hand＿model＿and＿multiple＿cameras

## INTECH

open science｜open minds

## InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83／A
51000 Rijeka，Croatia
Phone：＋385（51） 770447
Fax：＋385（51） 686166
www．intechopen．com

## InTech China

Unit 405，Office Block，Hotel Equatorial Shanghai
No．65，Yan An Road（West），Shanghai，200040，China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone：＋86－21－62489820
Fax：＋86－21－62489821
© 2007 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

