Recognition of Human Hand Motions for Robot Learning by Observation

Kiyoshi Hoshino, Takuya Kasahara, Motomasa Tomida, and Takanobu Tanimoto
(University of Tsukuba)

Abstract

The authors introduce you in this paper, not the method of imitation learning itself, but the method of hand pose estimation with single or dual high-speed camera. We believe that hand pose estimation at high speed and with high accuracy, i.e. recognition of human hand motions, is essential for robot learning by observation or its imitation learning. The purpose of this paper is to propose a remote-controlled robot system capable of accurate and high-speed performance of the same operation in strict conformity to the movement of the human operator, without sensors being installed or special control means being used. In particular, this paper intends to introduce a method for implementing a high-precision 3D finger pose estimation at a high speed that permits real time operation of a remote-controlled robot by two cameras installed at positions of loosely orthogonal relationship, using one PC of the normal specifications.

1. Introduction

The authors introduce you in this paper, not the method of imitation learning itself, but the method of hand pose estimation with single or dual high-speed camera. We believe that hand pose estimation at high speed and with high accuracy, i.e. recognition of human hand motions, is essential for robot learning by observation or its imitation learning.

The robot research and development projects implemented so far have not yet succeeded in incorporating a high level of intelligence in a robot. For example, when an object having various poses, weights and centers of gravity is located in front, it is not easy to ensure that an object is held by the robot hand appropriately in conformity to particular characteristics of each object so that the object can be manipulated. The level of the intelligence that can be built in a robot will be that of a six-year-old child, at best. However, the human communities are entering the age of a declining birthrate and aging population. Especially in the field of physical distribution and elder care, robots will be required to have an advanced level of intelligence.

What is important at present is a paradigm shift in thinking. To be more specific, it is not easy to incorporate an advanced level of intelligence in a robot in such a way that the robot will take care of the work of assortment. Assume, for example, that a human operator is stationed in a room different from that for the assortment worksite, and he watches a monitor to find out that the items to be sorted out are traveling on a belt conveyer. In response to the scene appearing on the monitor, the human operator moves his fingers and arms, and the robot located in a remote position performs the similar movement. If this is possible, comparatively complicated sorting work can be performed by the robot, without an advanced level of intelligence built in the robot. This does not require the robot to have a high level of intelligence. Rather, it is only required that the daily human action is performed through the monitor.

Hand tracking is not the robot vision technology required in this case. What is needed will be hand pose estimation. To be more specific, in the technique of hand tracking, the direction of hand movement and distance of hand movement are subjected to image analysis, and are assigned to the functions of the robot and information communications equipment. This can be compared to the cases where, if the human operator has performed action of "scissors" in a game of "scissors, paper, rock", the robot is made to perform operation A. If the human operator has performed action of "paper", the robot is made to perform operation B. The technique of hand tracking is embodied in a pointing device where the direction of hand movement and distance of hand movement are detected and employed to perform the required work. The robot is not manipulated by the daily action performed by the human operator. By contrast, in the technique of hand pose estimation, the "pose and posture of the hand" are associated with the dynamic behavior of the robot. To be more specific, in the

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

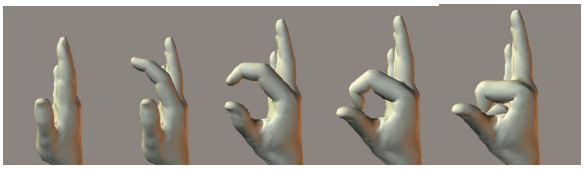
technique of hand pose estimation, the same movement as that of the human operator is re-configured by the robot. This does not required the user to learn a specific action in advance to ensure that the robot performs a specific function. If the user performs the daily action, the robot will do the same.

Two approaches are used to roughly classify conventional hand pose estimation - 3D-model-based and 2D-appearance- based. The 3D-model-based approach [1]-[6] involves extracting local characteristics, or silhouettes, in image recorded using a camera and fitting a 3D hand model constructed beforehand on a computer. While this approach estimates hand shapes highly accurately, it processes self-occlusion poorly and requires long processing time. The 2D-appearance-based approach [7]-[9] involves directly comparing an input image to an image stored in a database. While this approach reduces calculation time, if 3D changes in hand appearance - including wrist and forearm movements - are not an issue, this approach requires a large reference database and robot hand movement is difficult to control using imitation. If basic difficulty in estimating hand poses lies in hand shape complexity and self-occlusion, high-accuracy poses become theoretically possible to estimate, but this requires an extensive database including all possible hand images, including complexity and self-occlusion. The feasibility of this approach therefore depends on the search algorithm.

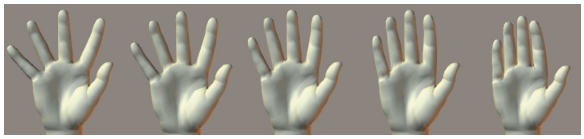
Regarding the 2D-appearance-based approach, Hoshino et al. proposed using computer graphics (CG) editing software and data gloves to create a large database containing personal hand pose attributes such as joint movable range and bone length [8]. They developed a search algorithm that shortens search time in looking for un-known input images by using a multi-layer database based on a self-organization map accompanying self-multiplication and self-extinction so that similar hand images are brought into closer proximity and by reducing the search area so that no data other than that near the search result during the previous search time will be inquired about [10].

In the technique of hand pose estimation using one camera, self-occlusion is fatal to the manipulation of an object by a remote-controlled robot. For example, assume that, when an object is photographed by the camera from the back of the hand, the silhouette is almost the same. In this case, there are at least two types of postures, such as power grasping and precise pinching. If the positional relationship is inaccurate between the finger and the object to be gripped, the object will easily get out of the robot hand. However, when consideration is given to an application example of the hand pose estimation technique, it is not realistic to use many cameras to photograph an object by surrounding the object to be manipulated, as in the case of the multi-camera system. If possible, the requirements should be met by installing two cameras at positions of loosely orthogonal relationship, without the camera installation position being specified in a precise manner.

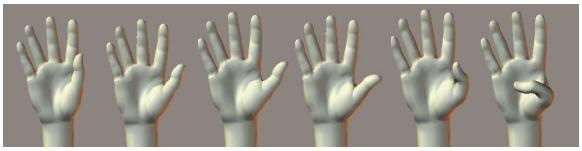
(a) Bending/extending of each finger.



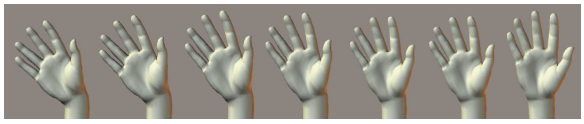
(b) Spreading out of the fingers.



(c) Thumb motion patterns.



(d) Wrist inclination examples.



(e) Forearm rotation.

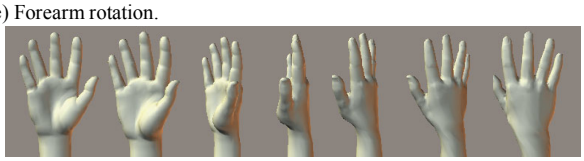


Figure 1. Additional hand poses derived from basic poses.

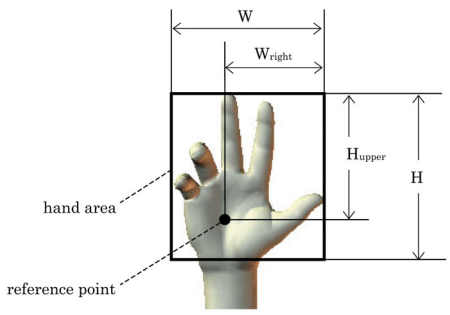


Figure 2. Proportional information of an image.

In view of the above-mentioned background, the purpose of this paper is to propose a remote-controlled robot system capable of accurate and high-speed performance of the same operation in strict conformity to the movement of the human operator, without sensors being installed or special control means being used. In particular, this paper intends to introduce a method for implementing a high-precision 3D finger pose estimation at a high speed that permits real time operation of a remote-controlled robot by two cameras installed at positions of loosely orthogonal relationship, using one PC of the normal specifications.

2. System Configuration

2.1 Data Sets

The database for our previous system was constructed using a single type of hand model, i.e. the experimenter's hand [12]-[13]. Namely, the database was prepared with storage by pairing individual hand images and finger and wrist angles synchronously acquired from a data glove and camera. Images were recorded using a camera at a resolution of 320*240 pixels laterally and vertically viewing hands and fingers on a big enough screen. Fingers and wrist angles were acquired using a data glove (Cyber Glove, Virtual Technologies Inc.), that obtained 18 types of angular information on the hand at a time. The database had about 30,000 datasets.

The database must contain every possible hand pose for that hand model, without exception. In this paper, we provide the system with two types of hand model pose patterns. In the discussion that follows, these two types of pattern are referred to as basic pose patterns and additional pose patterns. Both of them are generated using 3D computer graphics [8] (Poser 5, Curious Labs). The basic pose patterns are created to cover all hand poses. We independently captured images of the bending/extending of the index finger, middle finger, ring finger, and little finger in turn, the degree to which the fingers are spread out or close to one another, thumb motions, wrist motions, and the rotation of the forearm, and we saved data sets representing combinations of these poses to the database. For wrist motions, we only moved the wrist within the same plane, relative to the camera, for each rotation of the forearm.

Next, we used the additional pose patterns to add data sets for the poses when the palm or back of the hand is facing the camera. Whereas we had treated the extent to which the fingers are spread out from one another as one degree of freedom, actually the fingers are all capable of moving independently toward or away from the other fingers, and there is a great different in the appearance when the palm or back of the hand is facing the camera. Therefore, we added further hand pose data that combined the basic pose patterns for the thumb motions and wrist motions with new patterns for the bending/extending of each finger and the extent to which the fingers are spread out. Figure 1 shows examples of the additional bending/extending and spreading out of the fingers.

In this way, we constructed a database containing 772,576 data sets. This is roughly 25 times larger than the scale we were able to achieve using our previous method.

2.2 Calculation of Proportional Information on Hand Images Title

First, the contours of hand were defined. Specifically, the outermost pixel is given Labeling No. 1, and the pixel internally adjacent to the outermost pixel is given Labeling No. 2. Repeating this labeling process, the pixel position that is given the largest labeling no. is found. This is the reference point. Second, a hand range was defined and cutout. On the original image obtained by the previous paragraph, the top end, left end and right end of the hand image correspond to the top end, left end and right end of the hand's contour respectively. The bottom end of the hand image is at a height lower than the reference point by the distance to such a pixel on the outermost contour that is nearest from the reference point (the distance is defined by the number (N) of pixels).

For a hand image as cut out by Paragraph above, the following three different proportions, as shown in Figure 2, are calculated.

(1) Tallness: $R_{tall}[i] = H[i] / (H[i] + W[i])$,

(2) Top-heaviness: $R_{topheavy}[i] = H_{upper}[i] / H[i]$,

(3) Right-biasness: $R_{rightbiased}[i] = W_{right}[i] / W[i]$.

Where, H denotes the number of pixels measured in the vertical direction within the cutout. W denotes the number of pixels measured in the horizontal direction within the cutout. H_{upper} denotes the number of pixels located above the base point. W_{right} denotes the number of pixels in the right region of base point. Suffix i denotes the dataset number in the database.

These three proportions correspond roughly to forearm rotation, bending of thumb, and bending of four fingers other than thumb respectively. This image interpretation by proportional information is used for the first stage coarse screening.

2.3 Calculation of Image Features

In the present study, an image was divided into 64 sections in total - 8 x 8 each in the vertical and lateral directions - and the respective divided images were represented by numbers of dots. Therefore, a single image is described using the image features as a dot of 1 pattern x 64 divided sections.

2.4 Construction of Database

As discussed above, when one camera is used for photographing, various postures can be included when the appearance is the same as viewed from one direction. When the silhouette is the same as viewed from the back of the hand, there are various positional relationships of the thumb with respect to four other fingers. Taking advantage of the two high-speed cameras installed at positions of loosely orthogonal relationship, we would like to introduce the method for configuring the database for high-precision estimation of the positional relationship of the thumb with respect to other fingers. To be more specific, the data set of the database for matching has the following five forms of information: (i) finger joints angles as well as wrist angles (18+3 DOFs) which hand CG images were generated with, (ii and iii) proportional information of each image (3 DOFs) obtained from two cameras, and (iv and v) hand image features (64 DOFs) obtained from two cameras.

The following describes the basic concept of the method for hand pose estimation by two cameras installed at positions of loosely orthogonal relationship: In the first place, comparison is made of respective hand regions captured by two cameras, and the image having a greater area is determined. This is followed by the step of roughly narrowing the scope of choices, using the proportional hand image information on one of the images having been selected ((ii) or (iii)) alone. For simplicity, the first processing determines the approximate posture as viewed from the back of the hand. Secondly, high-definition matching of the degree of similarity (i.e., (v) or (iv)) is carried out by using only the image features obtained by the cameras installed at positions of loosely orthogonal relationship, out of the selected candidates. For brevity, using the image viewed from the lateral position, the second processing determines how far the finger is bent.

3. Hand Pose Estimation

3.1 Hand Area Extraction

To extract the user's hand area, we use the background subtraction method. Where the background image is relatively stable, it is sufficient to generate in advance a background model by averaging a number of image frames that do not include the hand area. However, in most cases, there is some fluctuation in the background due to blinking light fixtures, changes in sunlight, foliage moving in the wind, and shadows from moving objects. Therefore, many methods have been proposed for background models that take this sort of background fluctuation into account. These can be divided into two main types: one type of method constructs a background model in advance, and the other dynamically updates the background model. Compared to the former method, the latter allows stable extraction of the movement area where there is significant change in the background, by dynamically modeling the background. However, there are also problems with this type of modeling, including high computing costs and the need for a large-capacity memory to achieve high-speed processing.

In the interest of achieving high-speed processing, for this research we used the method of constructing a background model in advance, based on our assumption of use in an indoor environment, where there may be fluctuation due to the lighting but shadows from moving objects can be ignored. First, the image captured by the camera is express by the RGB colorimetric system. This system is however greatly affected by changes in brightness, due to the high correlation between the various values. Therefore, our system converts the image data from the RGB colorimetric system to the HSV colorimetric system that has a uniform color space, based on equations (1) and (2).

$$H = \begin{cases} 60 * \left(\dfrac{G-B}{\max - \min} \right), & if\ \max = R, \\ 60 * \left(2 + \dfrac{B-R}{\max - \min} \right), & if\ \max = G, \\ 60 * \left(4 + \dfrac{R-G}{\max - \min} \right), & if\ \max = B, \end{cases} \tag{1}$$

$$S = \frac{\max - \min}{\max}\ , \tag{2}$$

$$V = \max(R, G, B)\ , \tag{3}$$

$$f(x, y) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) . \tag{4}$$

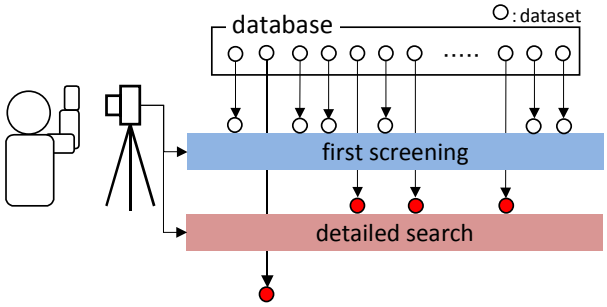


Figure 3. Two-step hand pose estimation method using two cameras installed at positions of loosely orthogonal relationship.

where, the *H*, *S*, and *V* in the equation stand for Hue, Saturation, and Value, respectively, of which our research uses *H* and *S*. The amount of fluctuation varies with the background area, but assuming that the fluctuations have a normal distribution, we use the background model shown in equation (4) for the *H* and *S* of each pixel. The μ and σ^2 in the equation represent the mean and the variance of the *H* and *S* in the N frame. When calculating the background difference to extract the hand area, if the difference for each pixel between the actual and the average *H* and *S* values is *k* times the standard deviation or less, the pixel is considered to be background.

Once the background and foreground have been separated using the background subtraction method, the system removes noise by means of morphological opening, and takes the maximum linked area of the foreground as the hand area.

3.2 Compensating for Forearm Inclination

For estimation, it is necessary that the user be able to move freely before the camera. In the images we used to construct our database, the hand and forearm appear from the bottom of the screen, but when estimating the system must be able to recognize hand poses regardless of the direction from which the hand appears. The proposed method uses the fact that inclination results in virtually no change to the outline of the forearm to calculate and compensate for forearm inclination.

First, the system looks for the four points: *S*, *S*', *E*, and *E*'. Points *S* and *E* are the pixels at which the outline of the hand and forearm area crosses the edge of the screen. The system traces the pixels that make up the outline of the hand from point *S* to point *E*, and calculates the inclination of each pixel. The inclination of each pixel is taken to be the inclination of a straight line linking it with two other pixels, located a few pixels before and after it on the outline. The next step is to calculate the standard deviation around each pixel. Where there is a significant change in inclination, the standard deviation is large, and where there is little change it is small. Where the standard deviation is above a certain threshold, the nearest point to *S* is taken as *S*' and the nearest point to *E* is taken as *E*'. The straight line connecting *S* and *S*' is called *L*s, and the straight line connecting *E* and *E*' is called L_E. The inclination of the forearm is taken to be the average of the inclinations of *L*s and L_E .

3.3 Two-Stage Search

The following shows the details of the first step of roughly narrowing the scope of choices, and the second step of high-definition matching of the degree of similarity: The first stage in a 2-stage search is coarse screening using proportional hand image information. The second stage is detailed screening for determining the image most similar among candidates selected in the first stage. The second stage uses similarity calculation based on specified image feature types. Figure 3 shows Two-step hand pose estimation method using two cameras installed at positions of loosely orthogonal relationship.

The first screening uses the three parameters defined by proportional information. If all three parameters fall within the specified threshold, the dataset is chosen as a candidate for the second screening. These three parameters and their thresholds are shown below.

(1) Tallness threshold: $Th_{tall} > \left| R_{tall}[i] - R_{current-tall} \right|$,

(2) Top-heaviness threshold:

$$Th_{topheavy} > \left| R_{topheavy}[i] - R_{current-topheavy} \right| ,$$

(3) Right-biased threshold:

$$Th_{rightbiased} > \left| R_{rightbiased}[i] - R_{current-rightbiased} \right| .$$

R_{tall}, $R_{topheavy}$, and $R_{rightbiased}$ are proportions representing tallness, top-heaviness, and right-bias of the hand image in the dataset under inquiry. $R_{current\text{-}tall}$, $R_{current\text{-}topheavy}$, and $R_{current\text{-}rightbiased}$ are proportions representing tallness, top-heaviness and right-bias of the current input image. Suffix *i* is the dataset number.

The second screening uses a Euclidean–distance-based similarity search to determine the highest possible image similarity. Dataset joint angles having the shortest distance among candidates chosen represent the result to be determined as the image having the highest possible similarity to the input image.

3.4 Arm Pose Estimation

The following describes how to estimate the upper limb attitude:

Firstly, photograph a checker board according to the Zhengyou Zhang procedure, and calculate the internal and external parameters of the camera. The internal parameters in this case are represented by the lens distortion, focal distance and projection offset in an image space. These internal parameters are calculated from the multiple checker board images captured by two cameras. The external parameters are represented by the position and rotation of the camera with respect to the world coordinate system. These external parameters are calculated from one set of checker board images captured by two cameras. The left top corner of one set of the images captured by two cameras indicates the origin of coordinates, which provides a basis for forming X-, Y- and Z-axes.

Secondly, estimate the bone position using a 2D real image with distortion. Contour of the arm is obtained by binarization and edge detection. In this case, assume that the edge of the arm is a straight line. When the arm is viewed from the side (hereinafter referred to as "upper camera"), the locus of the center of the inscribed circle

represents the centerline of the bone, and the radius of the inscribed circle at each position indicates the bone radius.

To be more specific, the row values on both ends of the edge are calculated in the specific direction of column in the coordinates (column and row) of a real image with lens distortion (where row 1 < row 2). A search is then made for a space where the edge point is located inside the radius of the cycle, with reference to radius (row2 - row1)/2. This is followed by the step of calculating the distance up to the edge within the range from row 1 to row 2. The minimum distance is recorded as an array with respect to each row value. After that, the row value where the distance is the maximum in the direction of row is taken as representing the position of the row. The above-mentioned distance in this case is assumed as representing the radium of the bone.

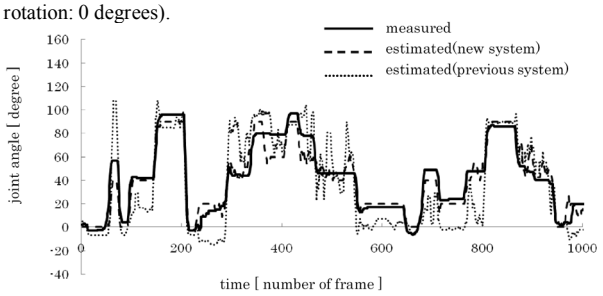
The above-mentioned calculation is performed for the upper camera and the image (hereinafter referred to as "lateral camera" of the arm as viewed from the top. It goes without saying that the relationship between the column and row is reversed for the lateral camera.

Thirdly, create the bone corresponding points, and recover the 3D position. In the first place, the lateral camera image is assumed as a reference image. The bone point sequence data of the lateral camera image and upper camera image is then converted into the bone point sequence data of the distortion-free space. Then, an epipolar line is obtained with respect to a point of the bone point sequence data of the lateral camera. In this case, the epipolar line is obtained by projecting the sight line determined by the punctual coordinates of the bone point sequence data, onto the distortion-free image space of the upper camera, using a camera parameter. The 3D recovery position is assumed as the position where there is a crossing between the sight lines at two points of the upper and lateral cameras.

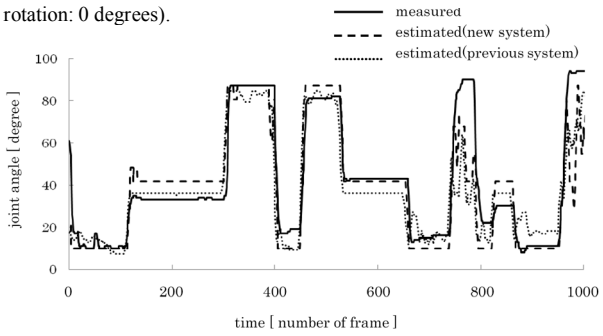
Fourthly, detect the wrist position and elbow position, and calculate the wrist and elbow vectors. In this case, the wrist position is found out as follows: The bone radius of the lateral camera at each bone position is multiplied by the corresponding bone radius of the upper camera, and the sectional area of the arm is obtained. A search is made by moving toward the wrist. If there is no updating for a prescribed distance, this is assumed as the minimum value, namely, the wrist position. The elbow position corresponds to the 3D position that conforms to the length from the obtained wrist position to the elbow having been inputted in advance.

To get the wrist vector and elbow vector, in the meantime, the covariance matrix of the 3D point sequence is found out within the length from the wrist to the elbow. The singular value of this matrix is analyzed. The corresponding to the eigen value providing the maximum singular value is considered as the vector from the wrist to the elbow (i.e., wrist vector). The vector from the elbow forward (i.e., elbow vector) can be obtained by the same processing, using the data from the obtained elbow position forward.

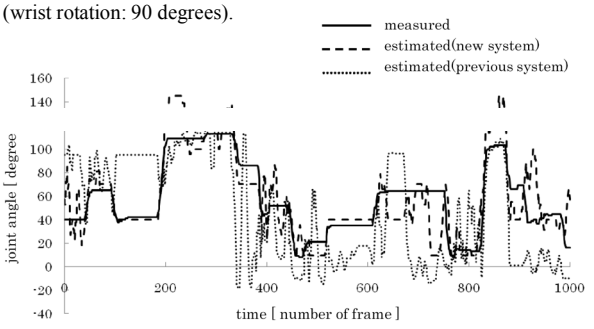
(a-1) PIP joint of index finger with palm facing the camera (wrist rotation: 0 degrees).



(a-2) CM joint of index finger with palm facing the camera (wrist rotation: 0 degrees).



(b-1) PIP joint of index finger with little finger facing the camera (wrist rotation: 90 degrees).



(b-2) CM joint of thumb with little finger facing the camera (wrist rotation: 90 degrees).

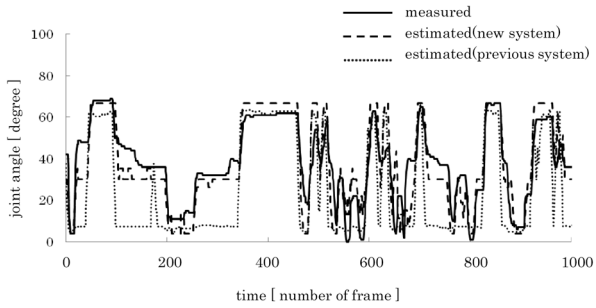


Figure 4. Examples of estimated results in subject M.T.

4. Estimation Experiment

4.1 Methods and Procedures

In order to verify the effectiveness of the system, the actual images were subjected to experimental estimation. Subjects held up a hand at a position approximately 1 m in front of the high-speed camera and moved the fingers and wrist freely. A motion of the hand was allowed in all the directions provided the hand was within the field angle of the camera. We employed a note PC (DELL Precision M4300, CoreTM 2 Duo Processor T8300 (2.40GHz, 800MHz FSB), main memory 4GB), a high-speed camera (Dragonfly Express™, Point Grey Research Inc.).

4.2 Results

For quantitative assessments, measured and estimated values must be compared, but in an ordinary environment using our approach, measured values of joint angle information from the hand and fingers moving in front of the camera cannot be obtained, so we conducted estimation experiments by making the same motions in both hands; one was recorded by the camera for hand pose estimation and the other wearing a data glove (Cyber Glove made by Virtual Technologies Inc.) for obtaining the joint angle. Subjects were instructed to move their hand and fingers freely in front of a high-speed camera.

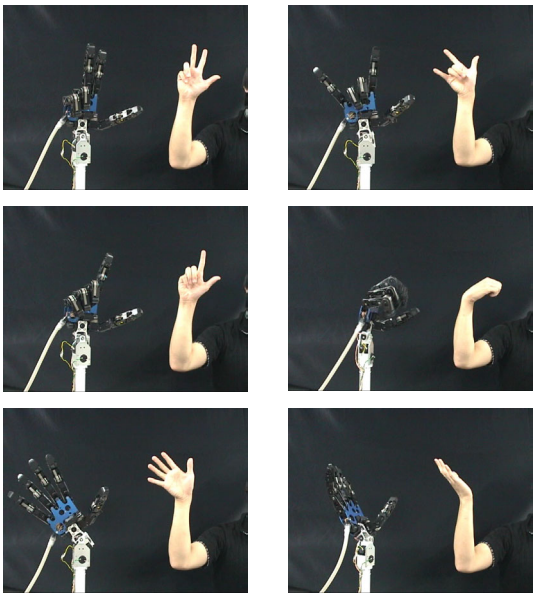


Figure 5. Snapshots of mimicking behavior as mobile manipulation.

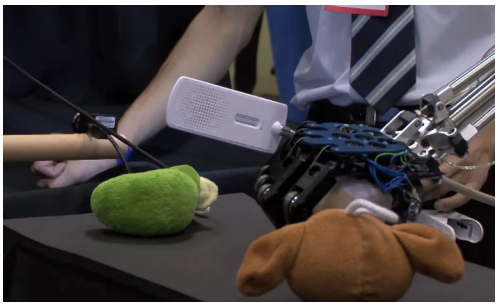


Figure 6. Remote control of a robot by hand pose estimation.

Results in Figures 4 show angular data measured using the data glove and estimated results by our previous system (about 30,000 datasets: See an example video at IJCAI2009 AI Video Awards [14]) and new system (about 750,000 datasets) in a subject. In these figures, (a) and (c), and (b) and (d) show the PIP joint of the index finger and CM joint of the thumb, when the palm is facing the high-speed camera and the little finger is facing the camera respectively. The state with the joint extended was set to 180 degrees. Mean and standard deviation scores of errors in estimated angles at index PIP were 0.45±14.57 by the proposed system, although 3.87±26.91 by our previous system, and at thumb CM, 4.7±10.82 and 9.5±15.77, respectively. Standard deviations of errors seem to be bypassed, but mean error is smaller, showing the improvement in accuracy even for specified users. The system operates at 80 fps using a notebook PC with a single high-speed camera and enables real-time estimation.

Figure 5 shows snapshots of the mobile manipulation by hand pose estimation technology. Figure 6 shows a snapshot of remote control of a humanoid robot by hand pose estimation system.

5. Discussion

For a 3D hand pose estimation system, the following conditions [13] must be met: (i) Hand pose estimation must be sufficiently accurate with a joint angle estimation error of a maximum of 4 to 5 degrees. (ii) Processing speed must be sufficiently high - at least 100 fps. (iii) All users must be processed, regardless of different hand size and shape. The approach in this paper meets these three conditions. Other conditions that should be additionally satisfied include: (iv) Relatively fast hand movement such as for sign language - possibly representing a user's natural movements - must be accepted, and (v) Both hands must be used, if possible.

We consider that the biggest reason for the improved accuracy with regard to unspecified users is the massive increase in the amount of data sets contained in the database. Because the proposed method constructs a database that includes all possible hand movements of a hand pose model, the number of data sets reached 772,576. In other words, a database that covers a single hand model evenly and in detail requires between several hundred thousand and several million data sets. However,

our previous method only created 30,000 data sets [8],[10],[12], a very small number. Under our previous method, a researcher created the database by wearing a data glove and forming various hand poses before the camera. While the researcher took care to cover all hand poses, the hand poses in the data base were influenced by the particular movements of the database creator. Therefore, a database created in this way inevitably relates to a single individual, which does not present a problem in estimating the hand poses of that individual, even if the database is small. However, we think that this did prevent the database from working well for unspecified users.

The biggest reason the previous database did not work well for unspecified users was that a person wearing a data globe could not simulate individual differences in the spreading apart of the four fingers. Because the spreading apart of the fingers involves the movement of the joint at the base of each finger, it greatly affects the appearance of the hand. However, it is difficult to cover all the possible combinations of the spreading apart motion and the bending/extending motion using a human hand as a model.

This paper has proposed the method for hand pose estimation using two cameras installed at positions of loosely orthogonal relationship. When distinction is to be made between similar operations such as grip holding and precision holding or stable gripping of an object is to be achieved, setting of the above-mentioned loose constraints will be permitted. However, if the hand pose estimation system is to be applied not only to the manipulation of a remote-controlled robot, but also to the information communications terminal by gesture, virtual key input device (so-called virtual key board), 3D-free pose input device, digital signage, finger motion capturing, it is preferred that pose estimation should be achieved successfully, using one or two camera images where the system side can be observed clearly, “even in the case of two through four cameras being installed appropriately” by each user. To be more specific, it is preferred that there should be no need of accurately specifying the position where a plurality of cameras are installed, unlike the multi-camera system, and the system should not use the camera position information. However, there is no knowing the image from which direction is fitted for pose estimation. Thus, it would be more preferred to provide a system that will provide “a plausible solution which is not very accurate in the strict sense of the word.” Solution to this problem will require a further study.

This research is funded by SCOPE project of MIC of Japan, and KDDI Foundation.

References

[1] Rehg, J.M., and Kanade, T. Visual tracking of high DOF articulated structures: an application to human hand tracking. European Conf. Computer Vision, pp.35-46, 1994.

[2] Jeong, M. H., Kuno, Y., Shimada, N., and Shirai, Y. Recognition of shape-changing hand gestures, IEICE Transactions on Information and Systems, E85-D, pp.1678-1687, 2002.

[3] Lu, S., Metaxas, D., Samaras, D., and Oliensis, J. Using multiple cues for hand tracking and model refinement. Proc. CVPR2003, 2, pp.443-450, 2003.

[4] Ueda, E., Matsumoto, Y., Imai, M., and Ogasawara, T. Hand pose estimation for vision-based human interface, IEEE Transactions on Industrial Electronics, 50, 4, pp.676-684, 2003.

[5] Jeong, M. H., Kuno, Y., Shimada, N., and Shirai, Y. Recognition of two-hand gestures using coupled switching linear model, IEICE Transactions on In-formation and Systems, E86-D, pp.1416-1425, 2003.

[6] Gumpp, T., Azad, P., Welke, K., Oztop, E., Dillmann, R., and Cheng, G. Unconstrained real-time markerless hand tracking for humanoid interaction. Proc. IEEE-RAS International Conference on Humanoid Robots, CD-ROM, 2006.

[7] Athitos, V., and Scarloff, S. An appearance-based framework for 3D hand shape classification and camera viewpoint estimation. Proc. Automatic Face and Gesture Recognition, pp.40-45, 2002.

[8] Hoshino, K. and Tanimoto, T. Real time search for similar hand images from database for robotic hand control. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E88-A, pp.2514-2520, 2005.

[9] Wu, Y., Lin, J., and Huang, T.S. Analyzing and capturing articulated hand motion in image sequences. IEEE Transactions on Pattern Analysis and Ma-chine Intelligence, 27, pp.1910-1922, 2005.

[10] Hoshino, K., Tamaki, E., and Tanimoto, E. Copycat hand - Robot hand imitating human motions at high speed and with high accuracy. Advanced Robotics, 21, pp.1743-1761, 2007.

[11] Otsu, N., and Kurita, T. A new scheme for practical, flexible and intelligent vision systems. Proc. IAPR. Workshop on Computer Vision, pp.431-435, 1998.

[12] Hoshino, K. and Tanimoto, T. Realtime hand posture estimation with Self-Organizing Map for stable robot control, IEICE Transactions on Information and Systems, E89-D, 6, pp.1813-1819, 2006.

[13] Hoshino, K. and Tomida, M.: 3D hand pose estimation using a single camera for unspecified users, Journal of Robotics and Mechatronics, 21, 6, pp.749-757, 2009.

[14] Hoshino Lab Web Page: http://www.kz.tsukuba.ac.jp /~hoshino/