https://doi.org/10.20965/jrm.2012.p0180

Paper:

Gesture-World Environment Technology for Mobile Manipulation – Remote Control System of a Robot with Hand Pose Estimation –

Kiyoshi Hoshino*, Takuya Kasahara*, Motomasa Tomida, and Takanobu Tanimoto***

*Graduate School of Systems and Information Engineering, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan
E-mail: hoshino@esys.tsukuba.ac.jp
**Crescent, Inc.
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan
[Received May 7, 2011; accepted August 31, 2011]

The purpose of this paper is to propose a remote-controlled robot system capable of accurate high-speed performance of the same operation strictly conforming to human operator movement without sensors or special control means. We specifically intend to implement high-precision high-speed 3D hand pose estimation enabling a remote-controlled robot to be operated using two cameras installed loosely orthogonally using one ordinary PC. The two cameras have their own database. Once sequential hand images are shot at high speed, the system starts selecting one database with bigger size of hand region in each recorded image. Coarse screening then proceeds based on proportional hand image information roughly corresponding to wrist rotation or thumb or finger extension. Finally, a detailed search is done for similarity among selected candidates. Experiments show that mean and standard deviation scores of errors in estimated angles at the proximal interphalangeal (PIP) index are 0.45 ± 14.57 and at the carpometacarpal (CM) thumb 4.7 ± 10.82, respectively, indicating it as a high-precision 3D hand pose estimation. Remote control of a robot with the proposed vision system shows high performance as well.

Keywords: 3D hand pose estimation, two cameras installed at position of loosely orthogonal relationship, 3D shape reconstruction of a hand from a 2D image, remote control of a robot

1. Introduction

Robot research and development projects have not yet succeeded in incorporating a high level of intelligence in a robot. When an object having various poses, weights and centers of gravity is located in front, for example, it remains difficult to ensure that the robot hand holds the object in conformance to individual object features so that the object can be manipulated. The level of intelligence built into a robot is currently that of a six-year-old child, at best. With countries such as Japan facing a declining birthrate and an aging population, robots are expected to be required to have an advanced level of intelligence especially in the fields of logistics and elder care meeting the needs and requirements of senior citizens.

A paradigm shift in thinking is needed. 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 in a room different from that of the assortment worksite monitors the area to confirm that items to be sorted travel as designated on a belt conveyer. In response to what is the operator's movements, a robot at the remote locations imitates the operator's movement. This would enable comparatively complex sorting without requiring that an advanced level of intelligence be built into the robot. This requires only that daily human action be done through a monitor.

Hand tracking is not the robot vision technology required in this case. What is needed is "hand pose estimation." Specifically, hand tracking in which images of hand movement direction and distance are analyzed and assigned to robot functions and information communication equipment. This is comparable to cases in which, for example, if the operator gestures "scissors" in a rock, paper, scissors game, the robot is made to do operation *A*. If the operator gestures "paper," the robot is made to do operation *B*. Hand tracking is enabled in a pointing device where hand direction and distance are detected and used to do the required work. The robot is not manipulated by daily operator action. Instead, in the technique of hand pose estimation, the "pose or posture of the hand" are associated with dynamic robot behavior. In hand pose estimation, the same movement as that of the operator is reconfigured by the robot. This does not require that the user to learn a specific action in advance to ensure that the robot does it. Once the user conducts a daily action, the robot will do the same.

Two approaches are used to roughly classify conventional hand pose estimation – 3-dimensional (3D) model-based and 2-dimensional (2D) appearance-based action. The 3D model-based approach [1–6] involves extracting local characteristics or silhouettes from images recorded by a camera and fitting a 3D hand model constructed beforehand on a computer. This approach estimates hand shapes highly accurately, but 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, which cuts calculation time. If 3D changes in hand appearance – e.g., wrist and forearm movement – are involved, however, this approach requires a large reference database, and robot hand movement is difficult to be controlled 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 of all possible hand images, including complexity and self-occlusion. The feasibility of this approach thus depends on the search algorithm.

In 2D appearance-based approach, Hoshino et al. [8] proposed using computer graphics (CG) editing software and data gloves to create a large database containing personal hand pose attributes such as movable joint range and bone length. They developed a search algorithm that shortens search time in looking for unknown 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 closer and the search area is reduced to only that data near the search result during previous search time is inquired about [10].

In hand pose estimation using one camera, self-occlusion is fatal to manipulating an object by a remote-controlled robot. Assume, for instance, that an object captured by the camera from the back of the hand has almost the same the silhouette. This may involve at least two types of postures, such as power grasping and precision pinching. If the positional relationship between the finger and object to be grasped or pinched is inaccurate, the robot hand will easily lose the object. When an application example of hand pose estimation is considered, however, it is unrealistic to use a multiple-camera system to capture an object by surrounding it. If possible, requirements should be met by installing two cameras positioned loosely orthogonally, without camera installation being specifically or precisely positioned.

Given the above background, we propose a remote-controlled robot system capable of accurate high-speed performance of the same operation strictly conforming to human operator movement, but without sensors or special controllers. We are particularly interested in introducing a way to implement high-precision high-speed 3D hand pose estimation enabling real-time operation by a remote-controlled robot using two cameras, positioned loosely orthogonally, together with an ordinary PC.

2. System Configuration

2.1. Data Sets

Our previous system database was constructed using a single hand model, i.e., the operator's hand [11, 12]. The database stored individual hand images paired with finger and wrist angles synchronously acquired from a data glove and camera. Images were recorded using a camera with a resolution of 320×240 pixels, laterally and vertically viewing hands and fingers on an appropriately sized screen. Fingers and wrist angles were acquired using a data glove (Cyber Glove, Virtual Technologies Inc.) that simultaneously obtained 18 types of angle information on the hand.

The database must contain every possible hand pose for a hand model, without exception. Here, we therefore provide a system with two types of hand model pose patterns – called basic and additional – generated using 3D computer graphics [8] (Poser 5, Curious Labs). Basic pose patterns are created to cover all hand poses. We independently captured images on bending and extending the index, middle, ring, and little fingers in turn, the degree to which fingers spread or close toward one another in five stages, thumb motions with six stages, and wrist motion and forearm rotation with seven stages. We saved data sets combining these poses in the database. Individual stages were decided based on dynamic range and joint DOF (degree of freedom) number. For wrist motions, we only moved the wrist within the same plane, relative to the camera, for each rotation of the forearm.

We used additional pose patterns to add data sets for poses when the palm or back of the hand faced the camera. Whereas we had treated how much the fingers spread mutually as one degree of freedom, fingers are actually all capable of moving independently toward or away from each other, so appearance when the palm or back of the hand is facing the camera differs greatly. We added further hand pose data combining basic pose patterns for thumb and wrist motions with new patterns for finger bending and extension and how much fingers spread. In other words, hand CGs with various poses are systematically generated through the former "basic pose" procedure, and hand CGs with individual differences are generated through the latter "additional pose" procedure. **Fig. 1** shows examples of additional bending/extending and spreading of fingers. The resulting database contained 772,576 data sets from collecting large-scale data sets.

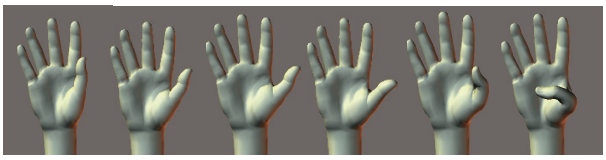
2.2. Calculation of Proportional Information on Hand Images

We first defined hand contours. Specifically, the outermost pixel becomes Labeling No.1 and the pixel internally adjacent to the outermost pixel Labeling No.2. Repeating this labeling yields the pixel position becoming the largest labeling found, i.e., the reference point. A hand range is defined and cut out. On the original image from the previous paragraph, the top, left and right ends of the hand image correspond to the top, left and right ends of

(a) Bending and extending fingers

(b) Spreading fingers

(c) Thumb motion patterns



(d) Wrist inclination examples

(e) Forearm rotation

Fig. 1. Additional hand poses derived from basic poses.

the hand contour. The bottom of the hand image is lower than the reference point in distance to such a pixel on the outermost contour nearest to the reference point – the distance is defined by pixel number (N).

For a hand image as cut out above, three proportions, shown in **Fig. 2**, are calculated.

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

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

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

H indicates the number of pixels measured vertically within the cutout. W indicates the number of pixels measured horizontally within the cutout. H_{upper} indicates the number of pixels located above the reference point. W_{right} indicates the number of pixels in the region to the right of reference point. Suffix i indicates the dataset number in the database.

These three proportions correspond roughly to forearm rotation, thumb bending, and nothumb finger bending. Image interpretation by proportional information is thus used for coarse stage-1 screening.

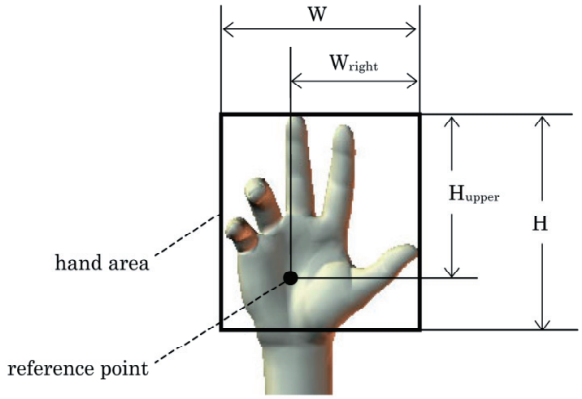


Fig. 2. Proportional image information.

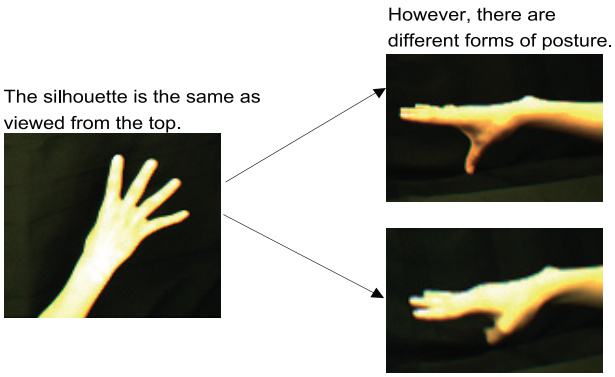


Fig. 3. Silhouette appearing the same in hand pose estimation by monocular camera but differing with types of posture.

2.3. Image Feature Calculation

In the present study, an image is divided into 64 sections – 8×8 each vertically and laterally – and divided images were represented by numbers of dots, i.e., M_0 pattern in HLAC [13]. A single hand image is thus described using image features as a dot of 1 pattern $\times$ 64 divided sections.

2.4. Database Construction

As stated, when one camera is used in capture, various postures can be included when the appearance is the same viewed from one direction, as shown in **Fig. 3**. When the silhouette is the same viewed from the back of the hand, various positional relationships of the thumb arise for the other four fingers. Taking advantage of the two high-speed cameras installed loosely orthogonally, we introduce a way for configuring the database for high-precision estimation of the positional relationship of the thumb to the fingers. Specifically, the data set of the database for matching has five types of information, shown in **Fig. 4** – (i) finger joints angles and wrist angles ($18+3$ DOFs) with which hand CG images were generated, (ii) and (iii) proportional information on each image (3 DOFs) obtained from two cameras, and (iv) and (v) hand image features (64 DOFs) obtained from two cameras.

The sections that follow describe basic concepts for

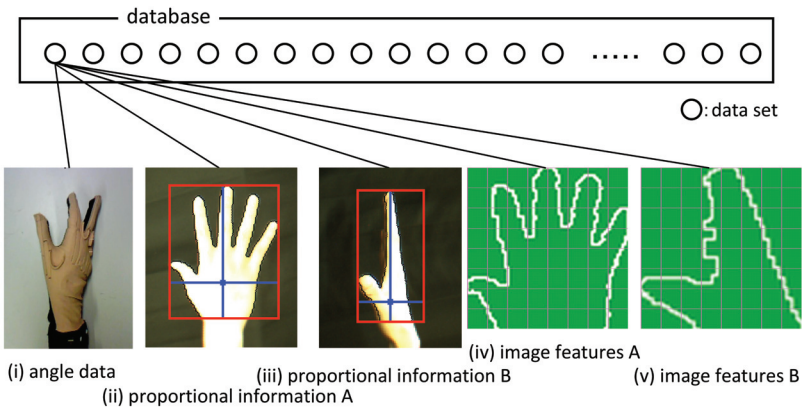


Fig. 4. Data set configuration in database for matching.

hand pose estimation using two cameras at positions loosely orthogonally. In stage 1, comparison is made of hand regions captured by two cameras, and the image having greater area is determined. The scope of choices is then roughly narrowed using proportional hand image information on one of the images selected ((ii) or (iii)) alone. For simplicity, the first processing determines the approximate posture viewed from the back of the hand. High-definition matching of the degree of similarity (i.e., (v) or (iv)) is done using only image features from cameras loosely orthogonally, from selected candidates. For brevity, using the image viewed from the lateral position, the second processing determines how far the finger is bent.

Because each data set has two types of image features as primitive learning data and paired with joint angle data, our system can allow 3D hand shapes to be reconstructed from 2D images.

3. Hand Pose Estimation

3.1. Hand Area Extraction

To extract the user's hand area, we use background subtraction. Where the background image is relatively stable, it is sufficient to generate a background model in advance by averaging a number of image frames that do not include hand area. In most cases, however, some fluctuation occurs in the background due to light fixtures blinking, sunlight changing, foliage moving, and shadows from moving objects. Many ways have thus been proposed for background models that consider such background fluctuation. These can be divided into two main types those for constructing a background model in advance and those for dynamically updating the background model. Compared to construction an advance background model, dynamically updating a background model enables more stable extraction of movement area where there is significant change in the background. This latter type of modeling has problems, including high computing cost and the need for large-capacity memory to ensure high-speed processing.

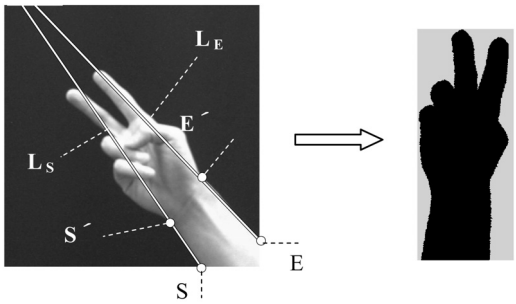


Fig. 5. Correction of forearm inclination.

To achieve high-speed processing, we constructed a background model in advance, assuming an indoor environment, where fluctuation may occur due to lighting but shadows from moving objects are ignored. An image captured by the camera is express by RGB colorimetrics. This is, however, greatly affected by changes in brightness due to the high correlation between various values, so our system converts image data from RGB colorimetrics to HSV colorimetrics having uniform color space.

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

3.2. Compensating for Forearm Inclination

Estimation requires that the user be able to move freely in front of the camera. In images used to construct our database, the hand and forearm appear from the bottom of the screen, but during estimation, the system must be able to recognize hand poses regardless of the direction from which the hand appears. We use the fact that inclination results in virtually no change to the forearm outline to calculate and compensate for forearm inclination.

The system first looks for four points S, S', E, and E' as shown in **Fig. 5**. Points S and E are pixels at which the hand outline and forearm cross the edge of the screen. The system traces pixels of the hand outline from point S to point E, and calculates individual pixel inclination. Individual pixel inclination is used as the inclination of a

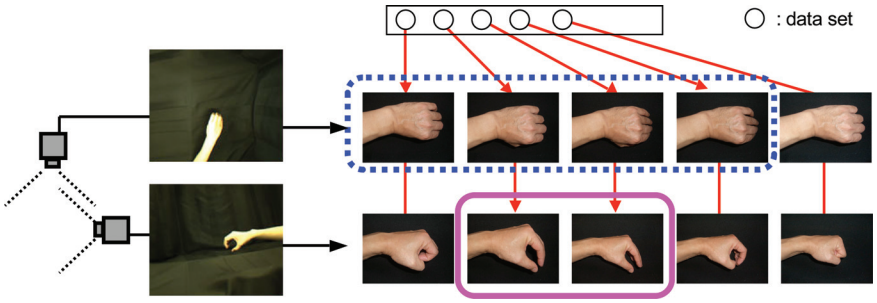


Fig. 6. Two-stage hand pose estimation using two cameras installed loosely orthogonally.

straight line linking it to two other pixels, located a few pixels in front of and after it on the outline. The next step is to calculate the standard deviation around each pixel. If inclination changes significantly, standard deviation is large, and vice versa. Where standard deviation exceeds a threshold, the nearest point to S is taken as S' and that nearest E as E'. The straight line connecting S and S' is called L_s and that connecting E and E' as L_E. Forearm inclination is used as the average of L_s and L_E inclination.

3.3. Two-Stage Search

In roughly narrowing the scope of choices and high-definition matching similarity, stage 1 in a 2-stage search involves coarse screening using proportional hand image information. Stage 2 is detailed screening determining the image most similar among candidates selected in stage 1. Stage 2 uses similarity calculation based on specified image feature types. **Fig. 6** shows two-stage hand pose estimation using two cameras installed loosely orthogonally.

Screening 1 uses three parameters defined by proportional information. If all three fall within the specified threshold, the dataset is chosen as a candidate for screening 2. These three parameters and their thresholds are shown below.

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

(2) Top-heaviness threshold:
$Th_{topheavy} > |R_{topheavy}[i] - R_{current\text{-}topheavy}|$,

(3) Right-biased threshold:
$Th_{rightbiased} > |R_{rightbiased}[i] - R_{current\text{-}rightbiased}|$.

R_{tall}, $R_{topheavy}$, and $R_{rightbiased}$ are proportions representing height, top-heaviness, and right-bias of the hand image in the data set in question. $R_{current\text{-}tall}$, $R_{current\text{-}topheavy}$, and $R_{current\text{-}rightbiased}$ are proportions representing height, top-heaviness, and right-bias of the current input image. Suffix i is the data set number.

Screening 2 uses a Euclidean-distance-based similarity search to determine the highest possible image similarity. Data set 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.

In stage 1, roughly narrowing the finger posture scope is done based on proportional information on the image having greater hand region. In stage 2, a high-definition finger posture is obtained from features of images in the loosely orthogonal relationship. The dotted line indicates step 1 of rough narrowing. The solid line indicates step 2 of high-definition matching of the degree of similarity.

3.4. Arm Pose Estimation

In estimating the upper limb attitude, we capture a checkerboard based on the Zhengyou Zhang [a] procedure, and calculate internal and external camera parameters. Internal parameters are indicated by lens distortion, focal distance, and projection offset in an image space. These internal parameters are calculated from multiple checkerboard images captured by two cameras. External parameters are indicated by camera position and rotation for world coordinates. These external parameters are calculated from one set of checkerboard images captured by two cameras. The left top corner of one set of these images indicates the origin of coordinates, which provides a basis for forming X-, Y- and Z-axes.

We estimate then bone position using a 2D real image with distortion. Arm contour is determined by binarization and edge detection, assuming that the arm edge is a straight line. When the arm is viewed from the side – "upper camera" – the locus of the center of the inscribed circle indicates the center line of the bone, and the radius of the inscribed circle at each position indicates bone radius.

Specifically, 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 circle referencing radius $(\text{row } 2 - \text{row } 1)/2$. This is followed by calculating the distance to the edge within the range from row 1 to row 2. The minimum distance is recorded as an array for each row value. The row value where the distance is maximum in the row direction indicates the row position. The above distance is assumed to indicate the bone radius.

The above calculation is done for the upper camera and the image – "lateral camera" – of the arm viewed from the top. The relationship between the column and row is reversed for the lateral camera.

We thirdly create points corresponding to bone, and recover the 3D position. The lateral camera image is as-

sumed as the reference image. Lateral camera image and upper camera image bone point sequence data is then converted to distortion-free space bone point sequence data. An epipolar line is obtained for a bone point sequence data point of the lateral camera by projecting the sight line determined by accurate coordinates of bone point sequence data onto the distortion-free image space of the upper camera using a camera parameter. The 3D recovery position is the position where sight lines cross at two points for the upper and lateral cameras.

We fourthly detect wrist and elbow positions and calculate wrist and elbow vectors. The wrist position is found 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 input in advance.

To get wrist and elbow vectors, the 3D point sequence covariance matrix is found within the length from the wrist to the elbow. This singular matrix value is analyzed. The corresponding to the eigenvalue providing the maximum singular value is the vector from the wrist to the elbow, i.e., wrist vector. The vector from the elbow toward the shoulder, i.e., elbow vector, is derived from the same processing, using data from obtained elbow position.

4. Estimation Experiments

4.1. Methods and Procedures

To verify the effectiveness of our proposal, actual images were subjected to experimental estimation. Subjects raised their hand to 1 m in front of a high-speed camera and moved their fingers and wrist freely. Hand movement was allowed in all directions, provided that it was within the camera field angle. We used a notebook PC (Dell Precision M4300, CoreTM 2 Duo Processor T8300 (2.40 GHz, 800 MHz FSB), main memory 4 GB) and a high-speed camera (Dragonfly Express^{TM}, Point Grey Research Inc.).

Technically speaking, the system works with one camera, but we used two here to enable the operator to handle various hand motions such as grasping and pinching and so that the remote robot could operate and handle an object appropriately and accurately based on the operator's motions.

4.2. Results

Figure 7 shows four examples of hand pose estimation in snapshots from two cameras positioned loosely orthogonally. Estimated results were drawn using the CG hand at the bottom of each example. Finger angles with wrist rotation were estimated with high precision, including different hand motions such as power grasping and precision pinching.

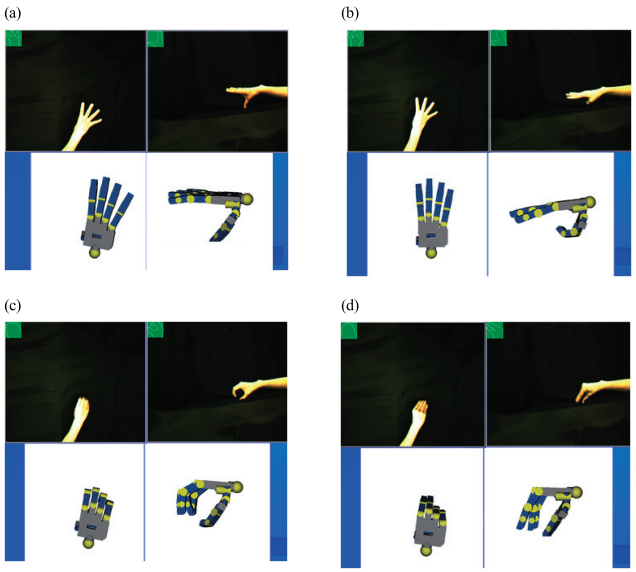


Fig. 7. Four types of captured hand images and results of hand pose estimation by two cameras installed loosely orthogonally. (c) Power grasping. (d) Precision pinching.

In quantitative assessment, measured and estimated data should be compared, but in an ordinary environment using the similar approach to ours, joint angle information data from the hand and fingers moving in front of the camera cannot be obtained. We therefore conducted estimation experiments making the same motions with both hands – one recorded by the camera for hand pose estimation and the other wearing a data glove (Cyber Glove, Virtual Technologies Inc.) – to obtain the joint angle. Subjects were instructed to move their hands and fingers freely in front of the high-speed camera.

Results in **Fig. 8** show angle data measured using the data glove and estimated results by our proposal with 750,000 data sets and our previous system. **Figs. 8(a-1)**, **(b-1)**, **(a-2)**, and **(b-2)** show the proximal interphalangeal (PIP) joint of the index finger and the carpometacarpal (CM) joint of the thumb, respectively, with the palm facing the high-speed camera or the little finger facing another camera. The state with the joint extended was set to 180°. Mean and standard deviation scores of errors in estimated angles at PIP index were 0.45 ± 14.57, and at thumb CM, 4.7 ± 10.82. Standard deviations of errors seem to be bypassed compared to our previous system [12], but mean error is lower, showing the improvement in accuracy. The system operates at 80 fps using a notebook PC with a single high-speed camera and enables real-time estimation.

Figures 9–11 show results for three subjects estimated with the new system, showing individual differences in estimation. The three have different hand shapes and motion. Subject M.T. is an experimenter and hand image parameters in the database were constructed based on his hand. Subject H.F. is an athletic man with large hands. Subject N.I. is a woman with small hands. Despite these differences, estimation results did not show big differ-

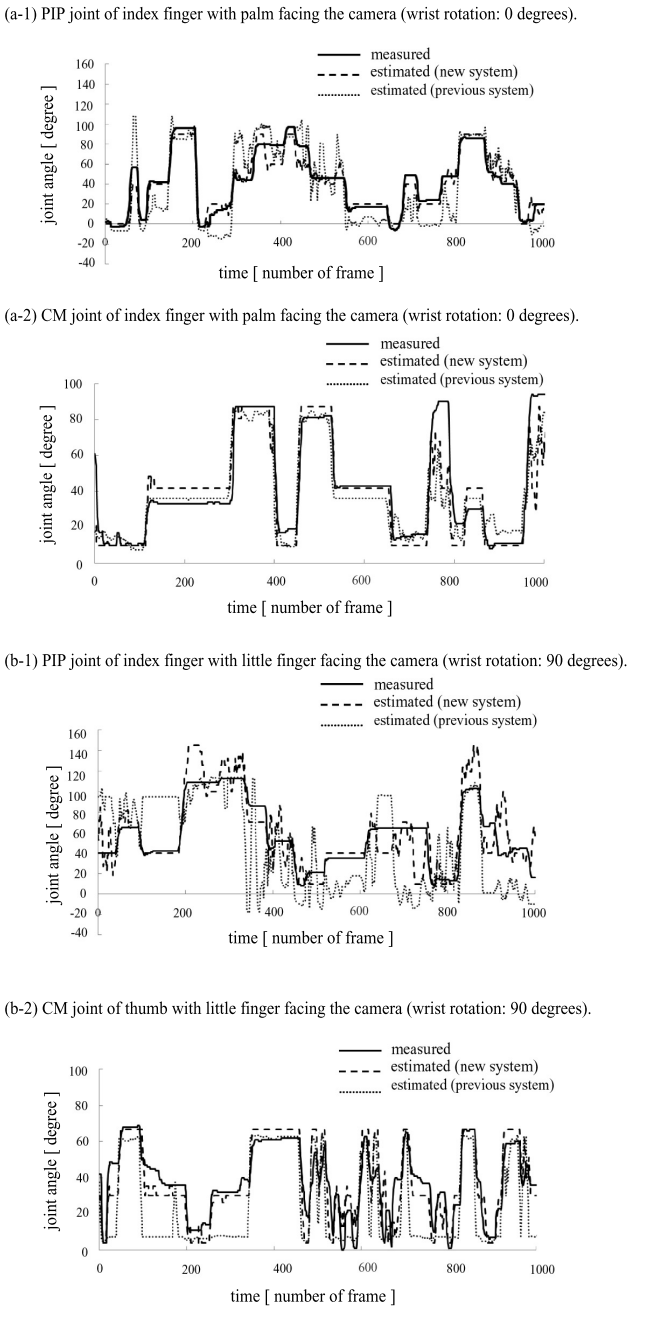


Fig. 8. Examples of estimated results for subject M.T.

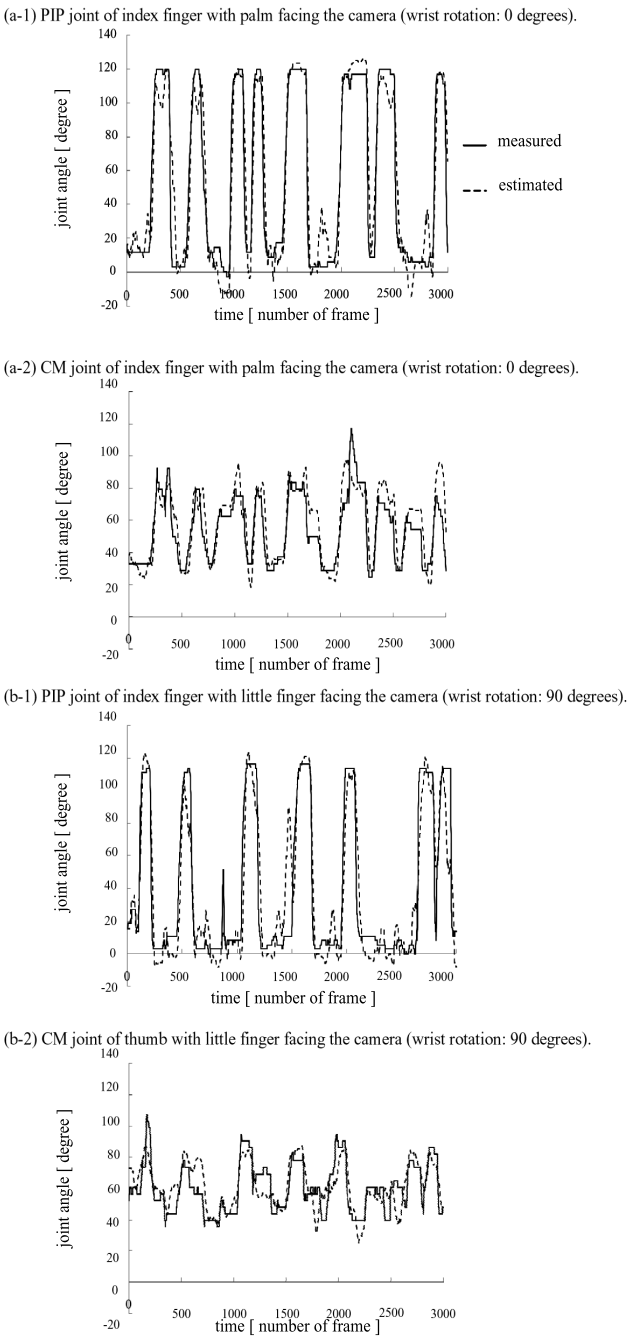


Fig. 9. Examples of estimation by subject M.T., with results drawn with new system and measured data.

ences.

Figure 12 shows robot remote control in hand pose estimation. (See a YouTube robot video clip [b], although a user put on a wrist band for arm pose estimation for exhibit.) Readers of this paper may see how the remote-controlled robot with our proposed vision technology works well.

5. Discussion

A 3D hand pose estimation system must fulfill the following conditions [12]:

(i) Hand pose estimation must be sufficiently accurate with joint angle estimation error at a maximum of 4° to 5°.

(ii) Processing must be sufficiently fast – at least 100 fps.

(iii) All users must be processed, regardless of different hand size and shape.

Our approach in this paper meets these three conditions. Other conditions that should also be satisfied include:

(iv) Relatively fast hand movement such as for sign language – possibly representing a user's natural movement – must be accepted.

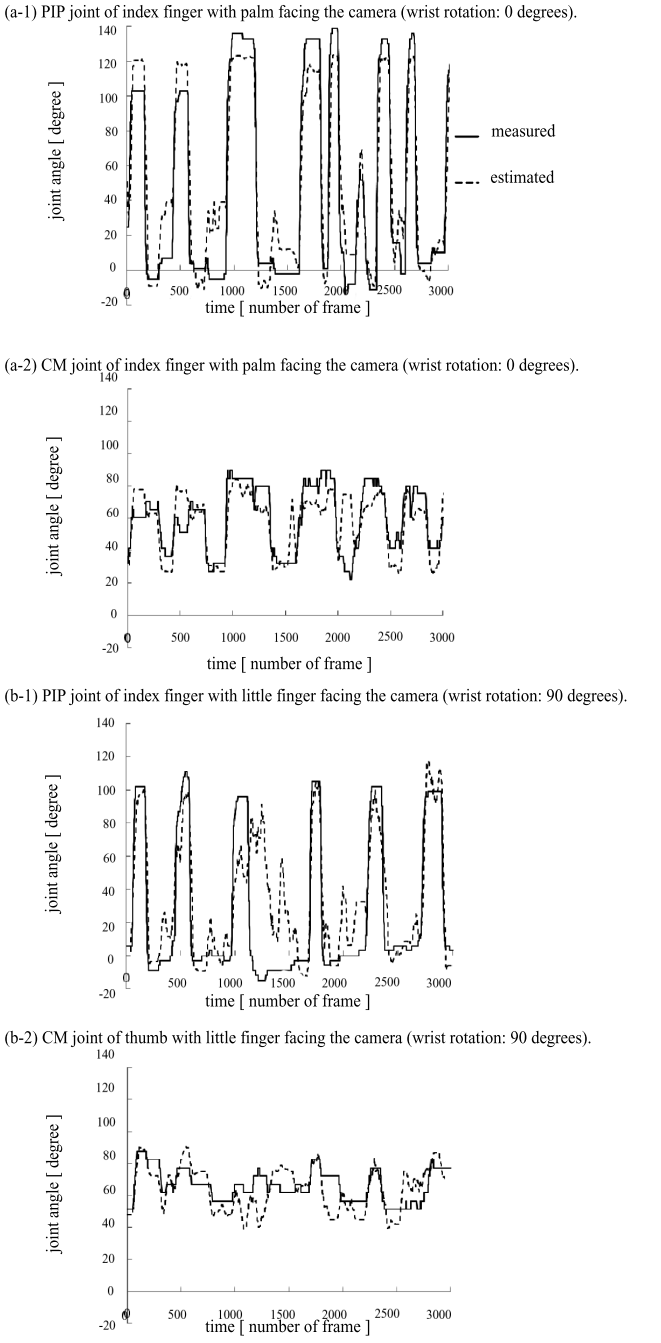


Fig. 10. Examples of estimation by subject H.F., whose hand images are not stored in the database.

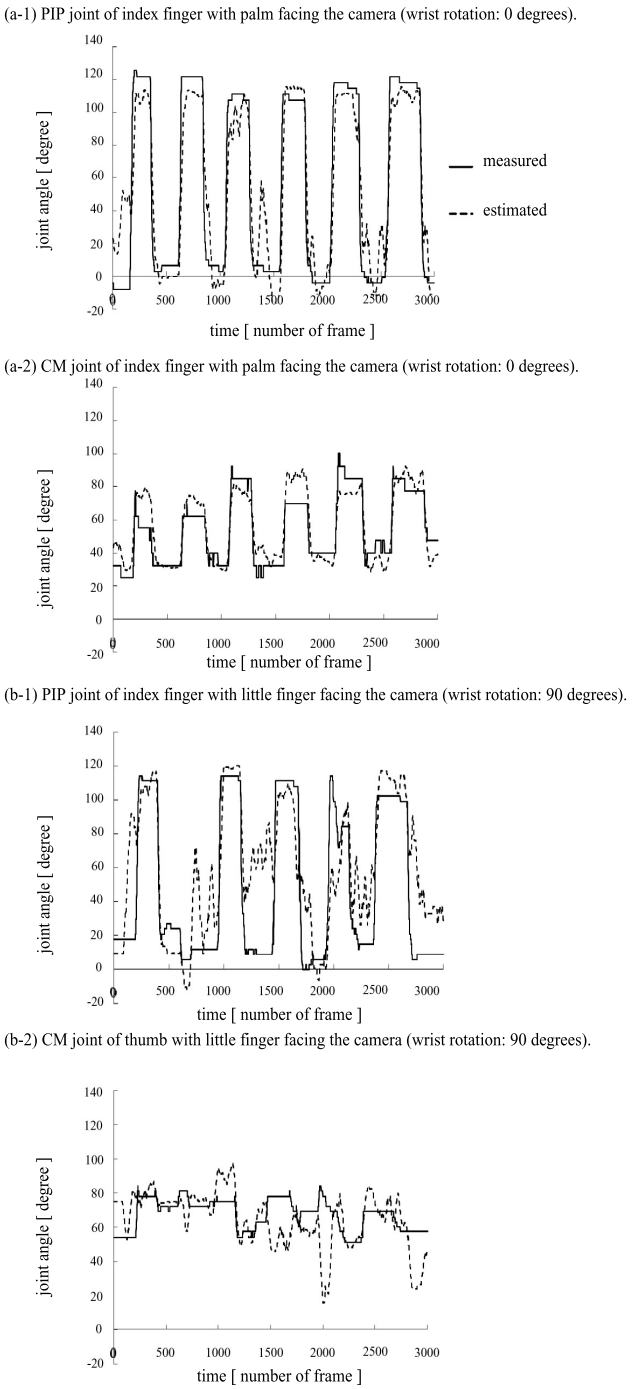


Fig. 11. Examples of estimation by subject N.I., whose hand images are not stored in the database.

(v) Both hands must be able to be used simultaneously, if possible.

Experimental results showed that thumb CM errors were 4.7 with a variation of 10.82. For condition (i), our results meet this condition, although variation was a little bit large. Variation mainly depends on database granularity, and we could decrease variation in output with adaptive filters such as Kalman or adaptive FIR and moving average, although output may produce a time delay depending on the filter time window. For condition (ii), our system did not satisfy processing speed. We used a notebook PC that accepts ExpressCard for the high-speed camera interface. The system may work at over 100 fps if it uses a more powerful PC. For condition (iii), **Figs. 9–11** in Section 4.2 show that our system satisfies the condition.

The biggest reason for improved accuracy for unspecified users is the massive increase in data sets in the database. Because our proposal constructs a database that includes all possible hand movements of a hand pose model, data sets number 772,576. These means that a database covering a single hand model evenly and in detail requires from several hundred thousand to several mil-

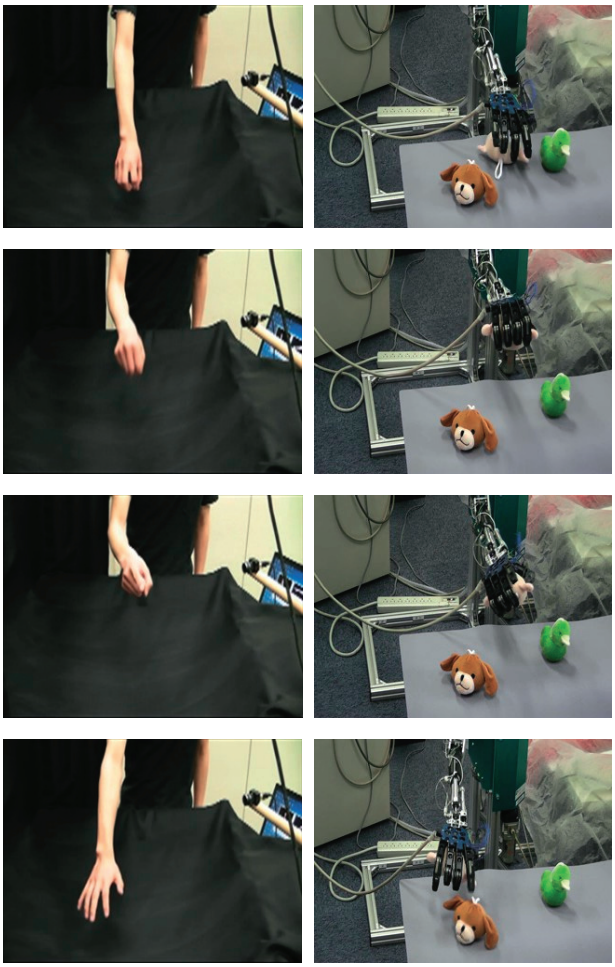


Fig. 12. Snapshots of robot remote control by hand pose estimation.

lion data sets. Our previous proposal created only 30,000 datasets [8, 10, 11]. In our previous method, a researcher created the database by using a data glove and forming various hand poses in front of the camera. While the researcher took care to cover "all" hand poses, hand poses in the database were influenced by the database creator's particular movements, so a database created this way inevitably reflects a single individual, which does not raise problems in estimating hand poses of individuals even if the database is small. We think, however, that this prevented 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 glove could not simulate individual differences in spreading the fingers because doing so involves moving the joint at the base of each finger, which greatly affects the hand's appearance. It is difficult to cover all possible combinations of such a movement and bending and extending the human hand as a model.

This paper has proposed hand pose estimation using two cameras installed loosely orthogonally. When a distinction is to be made between similar operations such as power grasping and precision pinching or an object is to be grasped stably, the above loose constraints are permissible, but if hand pose estimation is to be applied both to the manipulation of a remote-controlled robot and to information communication terminals by gesture, virtual key input – a "virtual" keyboard – 3D-free form input device, digital signage, or finger motion capturing, pose estimation should be achieved successfully using one or two camera images where the system can be observed clearly, "even using two to four appropriately installed cameras" by each user. Specifically, there should be no need to accurately specify where more than one camera should be installed, unlike the multicamera system, and the system should not use camera position information. There is no way of knowing the image from which direction is appropriate for pose estimation, however. Hand pose estimation using a 2D silhouette, for instance, is not robust for images observed and captured from the direction of the fingertip or wrist. If the number of cameras is limited, a user should install cameras where the system can observe the hands appropriately. The system should handle and overcome such problems automatically in the near future. 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." Solving this problem will require further study.

6. Conclusion

The purpose of this paper is to propose a remote-controlled robot system capable of accurate high-speed performance of the same operation in strict conformance to human operator movement, without sensors or special controllers. We specifically intended to introduce a method for implementing high-precision high-speed 3D hand pose estimation enabling real-time operation of a remote-controlled robot using two cameras installed loosely orthogonally and an ordinary notebook PC.

The two cameras have their own databases, with each database storing computer graphic hand images synchronously paired with fingers and wrist rotation angles. Each database has 772,576 data sets. Once sequential hand images are captured with high-speed cameras, the system selects a database with bigger hand regions in each recorded image. Coarse screening is based on proportional information on the hand image roughly corresponding to wrist rotation or thumb or finger extension. A detailed search then looks for similarity among selected candidates.

In system evaluation, we used a notebook PC having a Core™ 2 Duo Processor T8300 (2.40 GHz, 800 MHz FSB) and main 4 GB memory. Experiments showed that mean and standard deviation scores of errors in estimated angles at the index PIP are 0.45 ± 14.57 and at the thumb CM 4.7 ± 10.82. Processing time was 80 fps for hand pose estimation. Remote robot control with the proposed vision system showed high performance and experimental results indicated that our system enables high-precision 3D hand pose estimation at high speed.

Acknowledgements

This work was supported in part by the Japan Ministry of Internal Affairs and Communications (MIC), Strategic Information and Communications R&D Promotion Programme (SCOPE), and the KDDI Foundation.

References:

[1] J. M. Rehg and T. Kanade, "Visual tracking of high DOF articulated structures: an application to human hand tracking," European Conf. Computer Vision, pp. 35-46, 1994.
[2] M. H. Jeong, Y. Kuno, N. Shimada, and Y. Shirai, "Recognition of shape-changing hand gestures," IEICE Trans. on Information and Systems, Vol.E85-D, pp. 1678-1687, 2002.
[3] S. Lu, D. Metaxas, D. Samaras, and J. Oliensis, "Using multiple cues for hand tracking and model refinement," Proc. CVPR2003, Vol.2, pp. 443-450, 2003.
[4] E. Ueda, Y. Matsumoto, M. Imai, and T. Ogasawara, "Hand pose estimation for vision-based human interface," IEEE Trans. on Industrial Electronics, Vol.50, No.4, pp. 676-684, 2003.
[5] M. H. Jeong, Y. Kuno, N. Shimada, and Y. Shirai, "Recognition of two-hand gestures using coupled switching linear model," IEICE Trans. on In-formation and Systems, Vol.E86-D, pp. 1416-1425, 2003.
[6] T. Gumpp, P. Azad, K. Welke, E. Oztop, R. Dillmann, and G. Cheng, "Unconstrained real-time markerless hand tracking for humanoid interaction," Proc. IEEE-RAS Int. Conf. on Humanoid Robots, CD-ROM, 2006.
[7] V. Athitos and S. Scarloff, "An appearance-based framework for 3D hand shape classification and camera viewpoint estimation," Proc. Automatic Face and Gesture Recognition, pp. 40-45, 2002.
[8] K. Hoshino and T. Tanimoto, "Real time search for similar hand images from database for robotic hand control," IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E88-A, pp. 2514-2520, 2005.
[9] Y. Wu, J. Lin, and T. S. Huang, "Analyzing and capturing articulated hand motion in image sequences," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.27, pp. 1910-1922, 2005.
[10] K. Hoshino, E. Tamaki, and E. Tanimoto, "Copycat hand – Robot hand imitating human motions at high speed and with high accuracy," Advanced Robotics, Vol.21, pp. 1743-1761, 2007.
[11] K. Hoshino and T. Tanimoto, "Realtime hand posture estimation with Self-Organizing Map for stable robot control," IEICE Trans. on Information and Systems, Vol.E89-D, No.6, pp. 1813-1819, 2006.
[12] K. Hoshino and M. Tomida, "3D hand pose estimation using a single camera for unspecified users," J. of Robotics and Mechatronics, Vol.21, No.6, pp. 749-757, 2009.
[13] N. Otsu and T. Kurita, "A new scheme for practical, flexible and intelligent vision systems," Proc. IAPR. Workshop on Computer Vision, pp. 431-435, 1998.

Supporting Online Materials:

[a] Z. Zhang, "A Flexible New Technique for Camera Calibration," http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.35.6725&rep=rep1&type=pdf
[b] Gesture-Driven Robot Arm System, http://www.youtube.com/watch?v=UjbZYN1Db14

Name:
Kiyoshi Hoshino

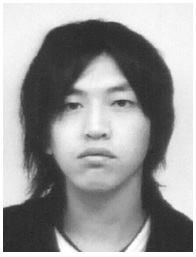
Affiliation:
Professor, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1993- Assistant Professor, Tokyo Medical and Dental University
1995- Associate Professor, University of the Ryukyus
1998-2001 Senior Researcher of PRESTO project, Japan Science and Technology Agency (JST)
2002- Associate Professor, University of Tsukuba
2002-2005 Project Leader of SORST project, JST
2008- Associate Professor, University of Tsukuba

Main Works:
• "Interpolation and extrapolation of repeated motions obtained with magnetic motion capture," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.E87-A, No.9, pp. 2401-2407, 2004.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• Japanese Society for Medical and Biological Engineering (JSMBE)

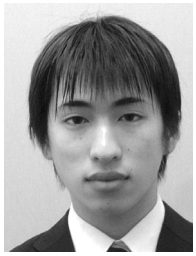


Name:
Takuya Kasahara

Affiliation:
Master Candidate, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2011 B.E. degree from University of Tsukuba
2011- Master candidate, University of Tsukuba



Name:
Motomasa Tomida

Affiliation:
Crescent, Inc.

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2008 B.E. degree from University of Tsukuba
2008- Ph.D. Candidate, University of Tsukuba
2011- Crescent, Inc.

Main Works:
• M. Tomida and K. Hoshino, "3D Hand Posture Estimation with Forearm Image Using Single Camera," The J. of the Institute of Image Information and Television Engineers, Vol.63, No.6, pp. 822-828, 2009.

Membership in Academic Societies:
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• The Virtual Reality of Society of Japan (VRSJ)

Name:
Takanobu Tanimoto

Affiliation:
Graduate School of Systems and Information Engineering, University of Tsukuba
Panasonic Corporation (Present affiliation)

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2004 B.E. degree from University of Tsukuba
2004- Master Candidate, University of Tsukuba
2009- Doctor Candidate, University of Tsukuba
2006- Panasonic Corporation

Membership in Academic Societies:
• The Institute of Electronics, Information and Communication Engineers (IEICE)