Development of a Human-Tracking Robot Using QR Code Recognition

Takashi Anezaki*, Koki Eimon*, Suriyon Tansuriyavong*, Yasushi Yagi**

* Okinawa National College of Technology.
905 Henoko, Nago City, Okinawa 905-2192

**The Institute of Scientific and Industrial Research,
Osaka University.
8-1 Mihogaoka, Ibaraki City, Osaka 567-0047

Abstract—**In this paper, we propose a human-tracking robot that can be used in commercial establishments such as airports and factories. The human-tracking process involves three main steps. The first is robust person identification using QR code recognition, and is the most important step in human tracking. The second is location detection using shape-based pattern matching in order to determine the position of the QR code when the person being tracked moves far from the robot. Finally, the third step is robot control to maintain an appropriate distance for human tracking. In experiments measuring QR code recognition rates, it was shown that QR code recognition was 99.9% and that location detection is robust. In a robot-control experiment, tracking was shown to be accurate, and during tracking, the robot maintained an appropriate distance from the human**

Keywords- Human -tracking robot, QR code, Shape-based pattern matching

I. INTRODUCTION

Industrial robots, which have been competing to do ever faster work within the confines of safety fences, are now being required to work cooperatively with humans, mingling with them outside of these fences[1]. Robots outside of safety fences must be capable of moving around safely, and to do so it is essential to be able to accurately teach them routes for movement.

In environments that include humans, robots must be made to understand everything about the environment in which they move, and as shown by the example of the Shakey robot from SRI[2], operating automatically in this way can be very expensive computationally. Also, environments with people are constantly changing, and understanding the environment including such changes makes it even more computationally expensive. The authors have conceived that it will be necessary for humans to show robots paths that are safe by walking through them, including in these ever-changing environments.

There are two possible ways a route can be taught to a robot by walking: either the human must lead the robot, or the robot must follow, or track, the person. In this paper, we discuss the case with the robot following the human. There is much need for human-following robots, for example, to carry baggage in commercial facilities such as airports or supermarkets[3].

There are three main requirements for realizing a human-following robot:

(a) Continuous identification of the individual person being followed.
(b) Detection of the location of the person being followed.
(c) Control of the robot based on the detected location data.

As such, we have proposed a human-following robot that uses a QR-code tag for tracking, considering that it would be difficult to satisfy condition (a) above using only image processing to implement following behavior.

II. HUMAN TRACKING USING QR CODES

QR codes have been designed to store a large amount of data, to be robust against distortion and damage, and to be readable at high speed and from all directions[4]. For this research, we focused on these features of QR codes, and implemented human tracking by equipping a three-wheeled robot with an image-processing system capable of recognizing QR codes. The system is designed for robust detection of individual persons by establishing a separate QR code corresponding to each robot. This system does not handle people specifically, but Katsuki et al. have proposed a system allowing robots to handle multiple heterogeneous objects by attaching 2D marks to the objects, such as QR codes, and encoding data specific to the objects[5] [6].

Earlier human-recognition systems have used technology such as active infrared tags[7]. These can be used to detect location easily for tracking purposes, but the tags require batteries, so the maintenance burden is large. In comparison, QR codes can simply be printed, creating inexpensive passive tags.

Although using QR codes for human tracking has the above advantages, the code is concentrated within a small frame, so depending on the camera performance, recognition rates can drop dramatically as the distance to the subject increases. Thus, to handle

QR codes that are farther removed, we propose a shape-based pattern matching method for detecting the location of the tag. We also considered that subjects could be concerned about wearing tags identifiable in visible light, so we studied use of an infrared camera with the QR codes printed in semitransparent retroreflective paint as well as badges with spherical retroreflectors on the four corners of the QR code. Retroreflective material is material which reflects incident light in the direction that it came from. Using spherical retroreflectors has the advantage that they can help locate the QR code when the camera anywhere at angles up to 90 degrees away from directly in front of the tag.

Of course, this does not cover cases when there is an occluding object between the user and the robot, and this will be a problem for future work.

III. SYSTEM ARCHITECTURE

A. System Architecture and Development Environment

The system configuration used for this research is shown in Figure 1. Robot control and image processing were developed using HALCON 9.0 from MVtec software[8] and run on the PC as shown in Figure 1. Aria from MobileRobots was also used for robot control[9].

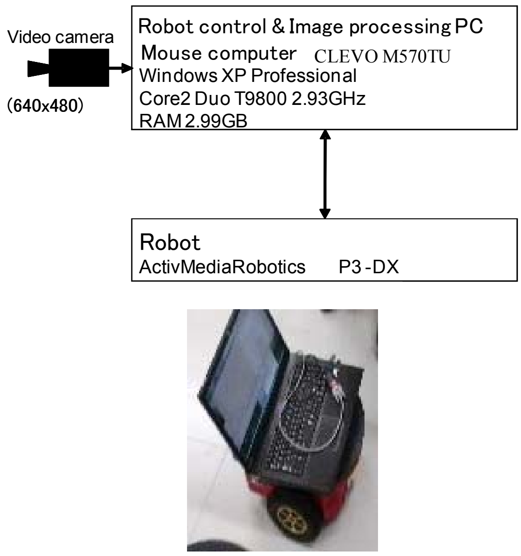


Fig. 1 System configurations

B. System Overview

The human-tracking robot operates according to the flow chart shown in Figure 2, and this is composed of the four processes indicated below.

The first is QR code recognition using a camera. This is the core

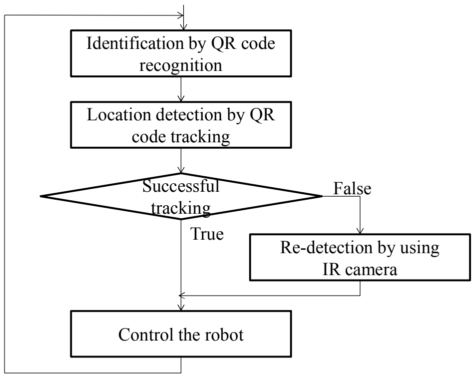


Fig. 2 Flowchart of tracking process

function for distinguishing between individual people. The second is the QR code tracking, which uses shape-based pattern matching to follow the subject (See Fig. 3). As discussed further below, the maximum distance for QR code recognition is short, and not sufficiently able to capture the marker continuously to be able to implement the following behavior. The QR code following function is designed to detect the location of the subject when it is more distant than the maximum distance for recognizing the QR code. The third process uses an infrared (IR) camera to relocate the tag when the QR code tracking function fails. For this research, we used a Nintendo Wii remote control[10] as an IR camera for preliminary experiments. The Wii remote control has an IR filter and CMOS sensor on the end, is able to capture an IR marker within its field of vision, and is easy to connect to a computer. The fourth process of the system determines robot motion based on the information obtained by the third process.

Fig. 3 Shape-based pattern matching

1) QR Code Recognition Function

For QR code recognition, a library included with HALCON 9.0 was used. In preliminary tests, we studied recognition rates for QR codes of area 42.25 cm^2 (6.5 cm x 6.5 cm) and 225 cm^2 (15 cm x 15 cm) for different distances (See Fig. 4). As a result, we found that for a 225 cm^2 mark, about the size of an A5 sheet of paper, we obtained recognition rates averaging 99.9% at distances from 50 to 160 cm. On the other hand, for a card size of 42.25 cm^2, recognition rates

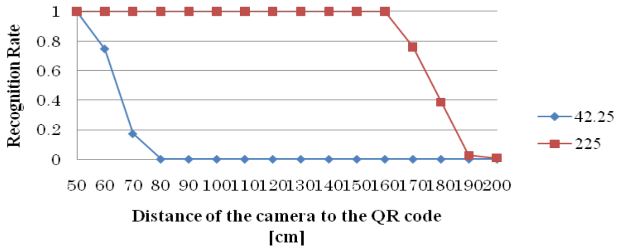


Fig. 4 Recognition rate of QR code

dropped at distances of 50 cm or more.

Recognition rates also drop dramatically when the camera position rotates more than 30 degrees on the yaw axis from directly facing the QR code, making it difficult to achieve stable recognition, but these cases can be handled using pattern matching together with an IR camera.

When causing the robot to follow, the authors decided that a distance of about 1 m to the user would be desirable. According to Nakajima[11], who studied the distances from mobile robots at which people began to feel uncomfortable (personal distance), even if the robot was only moving at 0.2 m/s, this personal distance was 0.95 m. We assmue that if the robot approaches closer than this distance for an extended period of time, the psychological stress on the user will begin to increase. Also, if the robot strays farther than this distance, there is room for obstructing objects to intervene, so about 1 m is appropriate as a following distance. In cases when the user begins walking quickly, it will be difficult to maintain a constant distance, and may not be possible to detect the position correctly. This type of case could be handled using a large-sized QR code, but to achieve more robust position detection, we proposed introducing shape-based pattern matching, to complement detection. In this way, QR code recognition is used to differentiate individual people, and position detection is implemented using pattern matching.

2) QR Code Tracking by Pattern Matching

Tracking with pattern matching is done using the procedure shown in Figure 5. First, at a time when the QR code can be recognized, the

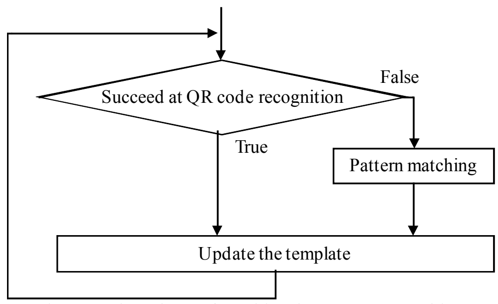
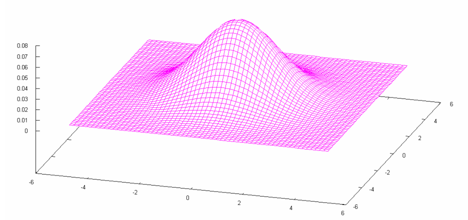


Fig. 5 Flowchart of tracking by pattern matching

image is smoothed using a Gaussian filter (See Fig. 6) to detect the



$$f(x,y)=\frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

$\sigma = 1.5$

$x, y : horizontal, vertical \quad element$

Fig. 6 Gaussian filter

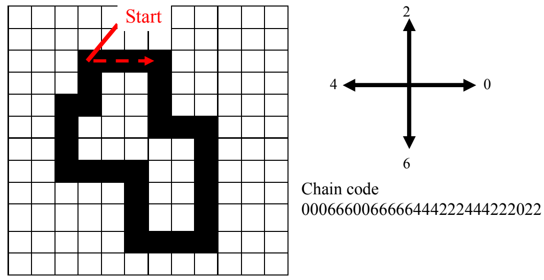


Fig. 7 Shape pattern by chain code

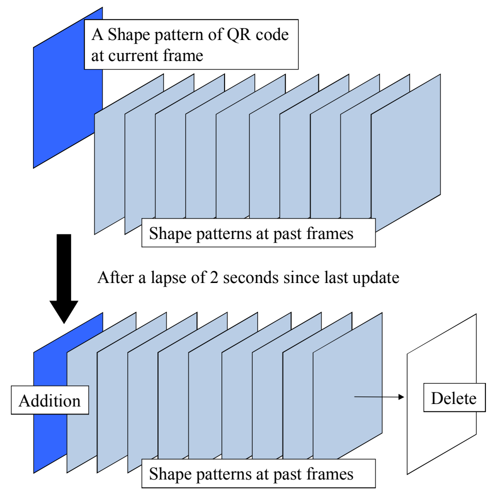


Fig. 8 Templates updating

outline of the QR code. Then, the extracted outline pattern is converted to a chain code, which is used as the template for pattern matching (See Fig. 7). The chain code has the advantages that it preserves the orientation of the outline as-is, it contains less information than pixel brightness and is not strongly affected by changes in image brightness. Processing to obtain this shape pattern is done using the HALCON 9.0 operator called create_scaled_shape_model.

When the QR code is too far from the camera and cannot be recognized, the shape pattern of the QR code, which has become small in area, is set as a template, and when the QR code is at a distance where recognition is possible, the template is updated with the shape pattern. Ten templates are kept from the previous 20 seconds, and when two seconds have elapsed since the latest template, a new template is added to the database and the oldest one is deleted (See Fig. 8). When performing matching, computation is done sequentially beginning with the most-recent template.

Five pyramid levels are used for the shape-based pattern matching (See Fig. 9). Level 1 is the original image, and images at half the size,

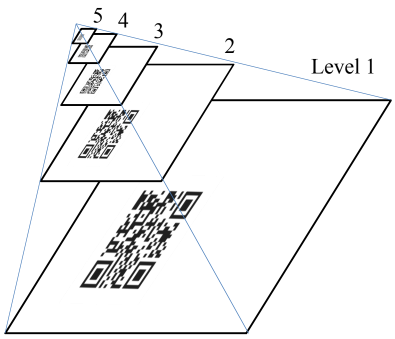


Fig. 9 Piramid levels in pattern matching

followed by ¼, 1/8, and 1/16 the size are generated for levels 2 to 5. Matching is done in order from the highest level, 5, which is the smallest image, reducing processing time. This shape-based pattern matching is accomplished using the find_scaled_shape_model operator in HALCON 9.0.

However, the camera is unable to recognize the QR code when it is blurred or is completely outside the field of vision, so position detection with an IR camera is needed when the position is unknown.

3) Robot Motion Control

The robot is controlled based on the location information obtained through pattern matching, in terms of two types of motion: rotation and forward/reverse motion. The amounts of each of these types of motion are determined by the following equations respectively.

$$R = R_T\left(1 - \frac{\text{QR code } x - coord}{C}\right) \quad (1)$$

$$F = F_T\left(1 - \frac{\text{QR code onscreen area}}{B}\right)$$
$$F = F_T\left(\frac{B}{\text{QR code onscreen area}} - 1\right) \quad (2)$$

Here, R_T is the maximum rotation value, defined as 30°, and C is the position at the center of the screen, which is 320 for our purposes. F_T is the maximum value for forward/reverse motion, set to 250 mm here. B is a reference value for the area of the QR code on the screen, used for deciding whether to move forward or in reverse. We define this to be 9400 pixels for the QR codes of area 225 cm^2. The relationship between QR code screen area and distance is shown in Figure 14. The upper part of equation (2) is for moving forward and the lower part is for moving in reverse. We experimentally determined the value of B such that it would equal the area of the QR code when the distance between robot and QR code is approximately 100 cm. The robot attempts to maintain this distance with its following behavior.

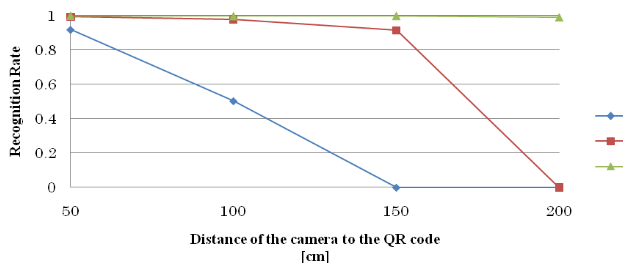
IV. Experiments and Discussion

A. Experiments in QR Code Tracking by Pattern Matchings

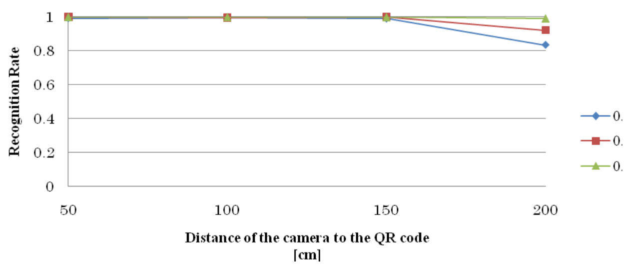
We performed tests to verify the behavior of QR code tracking using pattern matching, as discussed in *2)* of Ⅲ.

For the experiments, we set three levels, 0.9, 0.7, and 0.5, as the correlation lower bound when performing recognition with pattern matching, and then measured the recognition rates at distances of 50, 100, 150 and 200 cm from the QR code. The results for a QR code of area 42.25 cm^2 are shown in Figure 10 (a), and the results for a 225 cm^2 code are in (b). The measurements were made at a uniform brightness, with the QR code directly facing the line-of-site of the camera, and capturing an image at the appropriate distance. As a result, stable tracking was possible even with the 42.25 cm^2 code and setting the correlation lower-bound to the low value of 0.5, and even at distances of 200 cm. For the 225 cm^2 code case, we obtained favorable recognition rates even with a correlation lower-bound of 0.9.

Next, we conducted tests with a 42.25-cm^2 QR code, 1 m from the



(a) Area of QR code is 42.25cm^2



(b) Area of QR code is 225cm^2

Fig. 10 Recognition of QR code by using pattern matching

camera, and measuring recognition rates when the QR code was rotated away from perpendicular to the camera along the yaw, pitch, and roll axes (See Fig. 11). The results gave recognition rates nearing 80% with rotations up to 50 degrees on the yaw and pitch axes, confirming the robustness of QR codes to inclinations. Note that for all roll angles, QR codes were recognized with nearly 100% accuracy.

Also, once pattern matching had succeeded, we caused recognition

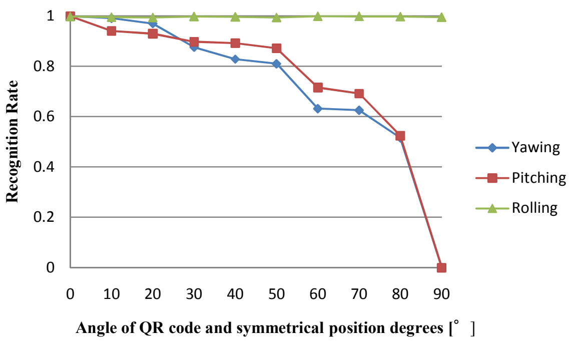


Fig. 11 Recognition of tilted QR code

to fail by moving the QR code violently, and even though the system lost sight of the code, we confirmed that it was able to re-discover it immediately (See Fig. 12). Additionally, when an occluding object came between the QR code and the camera during recognition, even though recognition failed temporarily, shape-based pattern matching was able to recognize the QR code after the obstructing object was removed.

B. Robot Motion Experiments

We also conducted experiments on the robot motion according to the method discussed in *3)* of Ⅲ. In the experiments, we travelled approximately 15 m around a table (4.5 m x 3 m) to confirm correct following behavior. We first attached a QR code to the person to be followed, and then approached the camera to a distance of about 50 cm to allow recognition. Then, we caused the robot to begin following by moving slowly backwards. In these experiments, we used a QR code of area 42.25 cm^2, did not include rediscovery using the IR camera, and only used shape-based pattern matching to performed following. The lower bound correlation value for pattern matching was set to 0.5.

Figure 13 shows the set path that the human walked and the resulting trajectory of the robot. Along the path, tape markers were placed only at the peripheral points, and the human subject walked using these as targets. The robot was made to follow, and odometry data while the robot was moving was used to generate the map.

When computing odometry, the path was divided up and calculated in sections. The odometry map is shown using a red line in the figure and was composed by piecing together the results. The width of deviation from the specified path was an average 6.07 cm, with the maximum being 17.4 cm. This was computed excluding the four corner sections. Figure 14 shows the changes in distance between the human and robot as the robot followed around the path. The right-

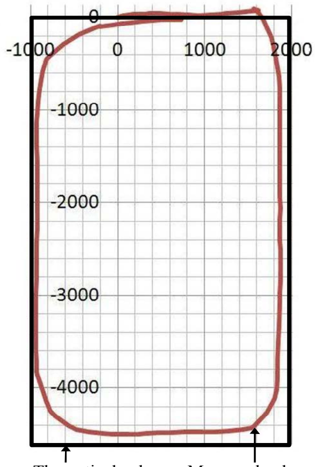


Fig. 13 Robot Trajectory

most point is the distance when tracking began, and the left point is the value at the end. In four locations on the diagram, the value increased dramatically. These are when the robot was performing a turn. Only a part of the QR code could be recognized due to image blurring, and this resulted in the distance calculated being too large. Besides these points, it is easy to see that the distance between camera and QR code was maintained between 100 and 150 cm while following the subject.

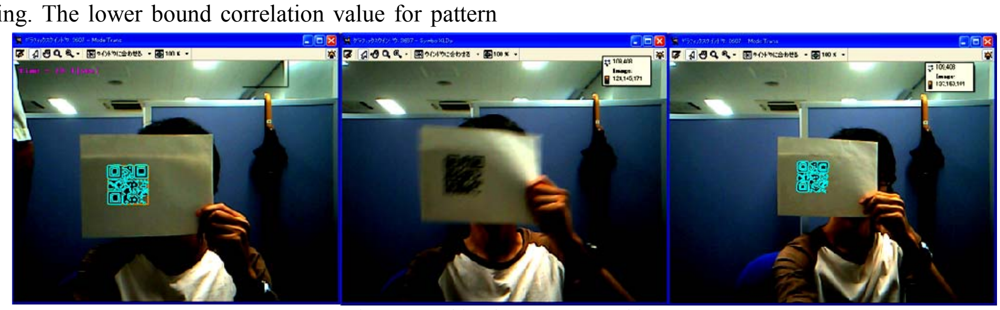


Fig. 12 Tracking by pattern matching
Left : QR code recognition, Middle : Failure recognition, Right : Recognition again by pattern matching.

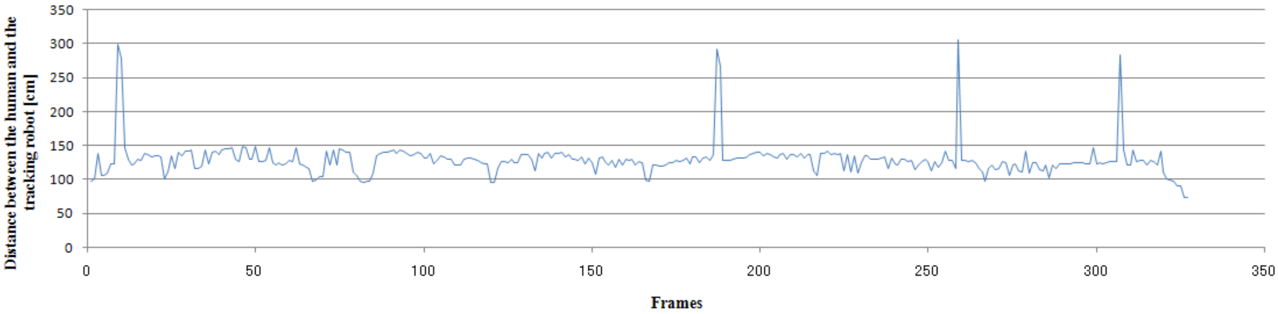


Fig. 14 Distance between the human and the tracking robot

C. *Discussion*

In the experiments with QR code tracking based on pattern matching in *A*, the template is continuously updated, so even if the QR code is shaken suddenly and moved so that it cannot be recognized, it can be re-discovered. We were also able to confirm that the method is robust against occluding objects.

We showed that following behavior was achieved in the robot motion experiments in *B*, but because R_T and F_T values, which limit any sudden motions, were set to fixed values in the experiments, the following motion was not able to keep up when the subject moved quickly. In order to resolve this issue, the speed of the subject being followed needs to be read, and the robot speed adjusted within a range that maintains stable motion. As well, if the subject being followed makes motions such as changing direction suddenly, it can be difficult to rediscover the QR code, so another future problem is to handle cases when the QR code is completely lost.

V. CONCLUSION

We have conducted research on human-tracking robots for use in work spaces such as factories or airports[13]. As the image processing system for this, we implemented a tracking system which differentiates individuals using QR codes and tracks subjects using shape-based pattern matching. We also proposed a rediscovery process using an IR camera to handle cases when the QR code strays completely out of the camera field of vision. The implementation consists of four components: shape-based pattern matching to track the QR code, an IR camera for rediscovery, and robot motion. With this implementation, we were able to confirm the robustness of the shape-based pattern matching against occlusions. Also, in tests using the IR camera with retroreflective materials, we demonstrated the effectiveness of the rediscovery process, but identifying the correct IR marker when there were several detected was a difficulty. In tests of the motion of the robot, it was able to follow while maintaining a set distance.

In the future, we plan to implement more-smooth robot motion and create a mechanism that can perform human-tracking in more practical environments.

REFERENCES

[1] Tomotada Saito: "Computing and Robotics Coexisting with Humans – Real-World Information Systems," Ohmsha Ltd. (2004-12) (Japanese).

[2] Nils J. Nillson: "SHAKEY THE ROBOT," SRI International, Technical Note 323, (1984-4).

[3] "The RoboPorter Intra-Airport Transport Robot Device Announcement and Demonstrations Conducted at Kitakyushu Airport," Kitakyushu Foundation for the Advancement of Industry Science and Technology Robot Development Support Office, News Release Feb. 7, 2008. (Japanese).

[4] Japan Automatic Identification Systems Association: "Everything you need to know about 2D barcode symbols," Ohmsha Ltd. (2004-2) (Japanese).

[5] R. Katsuki, J. Ota, Y. Tamura, T. Mizuta, T. Kito, T. Arai, T. Ueyama, T. Nishiyama "Handling of Multiple Heterogeneous Objects with Marks by a Robot," Trans. of the Japan Society of Mechanical Engineers. C, Vol. 70, No. 691, pp. 766-773 (2004-3) (Japanese).

[6] R. Katsuki, J. Ota, T. Arai, T. Ueyama, "Proposal of Artificial Mark to Measure 3D Pose by Monocular Vision," Trans. of the Japan Society of Mechanical Engineers. C, Vol. 71, No. 706, pp. 137-146 (2005-6) (Japanese).

[7] H. Aoki, S. Matsushita: "Balloon Tag: (In)visible Marker Which Tells Who's Who," Proc. of the 4th IEEE International Symposium on Wearable Computers, pp. 181-182 (2000-10).

[8] "HALCON 9.0.1 Released!" Lynx Corp., News Release, July 1, 2009 (Japanese).

[9] MobileRobots Inc., Aria, "http://robots.mobilerobots.com/wiki/ARIA"

[10] Nintendo Corp., Wii Remote, "http://www.nintendo.co.jp/wii/features/wii_remote.html".

[11] K. Nakashima: "Personal distance against mobile robot," The Japanese Journal of Ergonomics, Vol. 35, No. 2, pp. 87-95 (1999-4) (Japanese).

[12] Y. Nakazato, M. Kanbara, N. Yokoya: "Localization System Using Invisible Visual Markers and IR Camera for Wearable Augmented Reality," TVRSJ, Vol. 10, No. 3, pp. 295-304 (2005-9) (Japanese).

[13] Takashi Anezaki: "Proposal for Autonomous Navigation of Mobile Robot with Human-Copy Teaching for a 'Human-Friendly' Cell-Production Support Robot," IEEJ Trans. IA, Vol. 126, No. 11, pp. 1454-1459 (2006-11) (Japanese).