REAL-TIME REGISTRATION OF RGB-D IMAGE PAIR FOR SEE-THROUGH SYSTEM

Tatsuya Kittaka, Hiromitsu Fujii, Atsushi Yamashita and Hajime Asama

The University of Tokyo
Department of Precision Engineering, Faculty of Engineering
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

ABSTRACT

This paper presents a dense method of real-time registration of RGB-D image pair. So far, we have proposed the "see-through system", in which multiple images acquired from RGB-D sensors are integrated to present images that is useful for confirming the shape or the positions of objects behind obstacles. However, errors of positional relation of sensors result in see-through images in which some objects are doubled or appear at incorrect positions. It is difficult to align the images because they are captured from distant viewpoints and there are few shared field of view. In the proposed method, positional relation of two RGB-D sensors is corrected using a new IRLS framework, fast and robust minimization strategy.

Index Terms— diminished reality, RGB-D sensor, see-through, registration, remote operation

1. INTRODUCTION

For remote operation of robots in dangerous situations such as disaster sites, providing robot operators with appropriate visual information is important [1]. Since conducting tasks while comparing multiple images requires highly skilled technique [2], it is effective to integrate multiple images into one image which helps operators to recognize the environment. So far, we have proposed a real-time see-through system [3], in which images from RGB-D sensors mounted on a robot are integrated to produce see-through images, as illustrated in Fig. 1. In [3], RGB-D sensors are mounted in front of the robot and on the arm, and the sensor can move along with the arm to provide appropriate field of view.

However, there is a problem of "position gap", in which some objects are doubled or appear at incorrect positions in see-through images, because the information of the positional relation between sensors has small errors. If the gap exists, it is difficult to recognize the positions and shapes of objects in the image. This is a general problem to be solved, for not only this system but also other see-through applications [4–9].

In this paper, we propose a real-time registration method to correct positional relation of two RGB-D sensors to minimize the gap in see-through images.

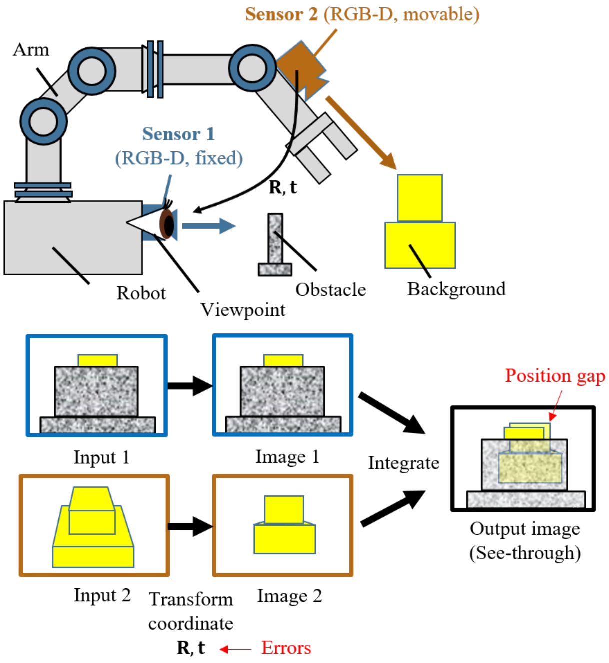


Fig. 1: The illustration of real-time see-through system for remote control robot using two RGB-D sensors [3]. However, there is a problem of small gap, caused by errors of positional relation of sensors, which leads to doubled objects in see-through images.

2. RELATED RESEARCH

Registration is related to visual odometry [10] or visual SLAM [11]. These techniques estimate sensor trajectory, that is, rotation and translation between time t and $t+1$ using two-dimensional images or three-dimensional data. As far as three-dimensional data are concerned, visual odometry and image registration are almost the same techniques, because appropriate registration of two set of data can be performed by applying appropriate rotation and translation.

This work was in part funded by ImPACT Program of Council for Science, Technology and Innovation (Cabinet Office, Government of Japan).

According to [12], visual odometry can be largely categorized into two branches. One is feature based method [13], which extracts features from images and matches them. Although the calculation cost of feature based methods is generally low, they can fail when appropriate features are not found or feature matching goes wrong. Therefore, we employ the other one, dense method, which directly matches values of each pixel in the images [10, 11, 14, 15].

However, there are two issues we have to solve when it is applied to see-through system for remote control robot:

1. Unlike visual odometry, see-through images are created by a pair of images captured from distant viewpoints. Thus the pair of images may have fewer shared field of view (the yellow part in the image 1 and 2 in Fig. 1, for example). It is difficult to align such images, because pixels out of shared field of view are supposed to have different colors and we have no information of where shared field of view is in unknown environment.

2. It is necessary to reduce calculation cost. Although dense method is robuster than feature based method, calculation cost is higher. For operation of remote control robot, real-time image presentation is required.

3. PROPOSED METHOD

The main concept of the proposed method is illustrated in Fig. 2. In this paper, we define the Sensor 1 and 2 as the RGB-D sensors, the Input 1 and 2 as images captured from respective sensors, and the Image 1 and 2 as the Input 1 and 2 transformed into the same coordinate system. The pose of the Sensor 2 (6 degree of freedom) is corrected and the appearance of the Image 2 is changed so that the difference between the Image 1 and 2 is minimized. If we define a six-dimensional vector $\boldsymbol{\xi}$ as the correction amount, the image registration is reduced to calculation of $\boldsymbol{\xi}_{\text{opt}}$, which minimizes an error fucntion $E(\boldsymbol{\xi})$:

$$\boldsymbol{\xi}_{\text{opt}} = \arg\min_{\boldsymbol{\xi}} E(\boldsymbol{\xi}). \tag{1}$$

In order to solve the two issues listed in Chapter 2, we have two new ideas:

1. Error function is minimized using a new Iteratively Reweighted Least Squares (IRLS) framework to enhance robustness against non-shared field of view.

2. Pixels of less importance are efficiently excluded using the gradient of images to reduce calculation cost.

3.1. See-through Images and Correction Amount

The method to create see-through images is explained in detail in [3]. The positional relation between the Sensor 1 and 2 are calibrated in advance. For each frame, three-dimensional point clouds captured from the Sensor 1 and 2 are coordinate-transformed to the coordinate system of the viewpoint of the output image, and then projected on two-dimensional images, creating the Image 1 and 2. The see-through image is created by alpha blending of the Image 1 and 2.

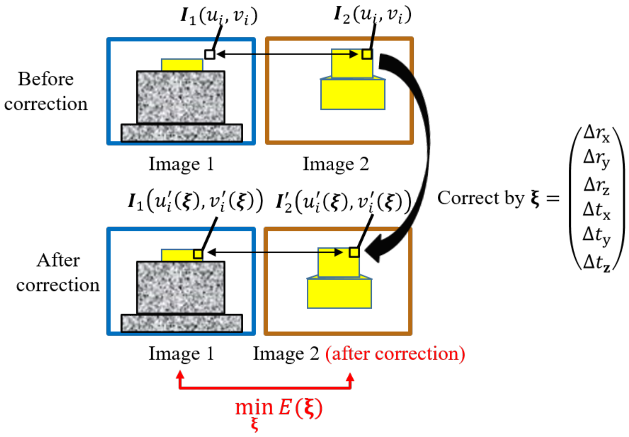

$\boldsymbol{I}_1(u_i, v_i)$
$\boldsymbol{I}_2(u_i, v_i)$
Before correction
Image 1
Image 2
$\boldsymbol{I}_1(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi}))$
$\boldsymbol{I}'_2(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi}))$
Correct by $\boldsymbol{\xi} = \begin{pmatrix} \Delta r_x \\ \Delta r_y \\ \Delta r_z \\ \Delta t_x \\ \Delta t_y \\ \Delta t_z \end{pmatrix}$
After correction
Image 1
Image 2 (after correction)
$\min_{\boldsymbol{\xi}} E(\boldsymbol{\xi})$

Fig. 2: The main concept of the image registration based on dense method. If we define a six-dimensional vector $\boldsymbol{\xi}$ as the correction amount, the image registration is reduced to minimization of an error function $E(\boldsymbol{\xi})$.

The focus of this research is errors of coordinate transformation, that is, rotation and translation. The errors are caused by motion of sensors. Rotation and translation can be represented as a six-dimensional vector $(r_{\text{x}}, r_{\text{y}}, r_{\text{z}}, t_{\text{x}}, t_{\text{y}}, t_{\text{z}})^{\text{T}}$, and correction amount of this vector can be represented as $\boldsymbol{\xi} = (\Delta r_{\text{x}}, \Delta r_{\text{y}}, \Delta r_{\text{z}}, \Delta t_{\text{x}}, \Delta t_{\text{y}}, \Delta t_{\text{z}})^{\text{T}}$. Image registration is reduced to calculation of the appropriate correction amount $\boldsymbol{\xi}$.

3.2. Derivation of Error Function

We derive the error function $E(\boldsymbol{\xi})$ on the assumption that corresponding pixels have the same values in the pair of images if they are correctly aligned. This assumption works as long as there are no strong specular reflection or semitransparent objects.

If the Image 1 and 2 have n pixels each, $E(\boldsymbol{\xi})$ is represented using pixel values of i-th pixel of the Image 1 and 2, for $i = 1, ..., n$. Let values of the i-th pixel (u_i, v_i) in the Image 1 and 2 be $\boldsymbol{I}_1(u_i, v_i)$ and $\boldsymbol{I}_2(u_i, v_i)$, respectively. Pixel values consist of three color components, R, G and B, so $\boldsymbol{I}_1(u_i, v_i) = (R_1(u_i, v_i), G_1(u_i, v_i), B_1(u_i, v_i))^{\text{T}}$, and the same is true of $\boldsymbol{I}_2(u_i, v_i)$. After correction by $\boldsymbol{\xi}$, the pixel (u_i, v_i) in the Image 2 goes to $(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi}))$ and the Image 2 changes to $\boldsymbol{I}'_2(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi}))$. Considering color consistency of the Image 2, illustrated as the curved arrow in Fig. 2, we define the residual of i-th pixel as follows:

$$\begin{aligned} r_i(\boldsymbol{\xi}) &= \boldsymbol{I}_1(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi})) - \boldsymbol{I}'_2(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi})) \\ &= \boldsymbol{I}_1(u'_i(\boldsymbol{\xi}), v'_i(\boldsymbol{\xi})) - \boldsymbol{I}_2(u_i, v_i). \end{aligned} \tag{2}$$

Using the residuals, we define the error function $E(\boldsymbol{\xi})$ as the sum of weighted quadratic form:

$$E(\boldsymbol{\xi}) = \sum_{i\in\mathcal{S}} w_i \boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{r}}^{-1} \boldsymbol{r}_i(\boldsymbol{\xi}). \tag{3}$$

$\boldsymbol{\Sigma}_{\mathrm{r}}$ is a 3×3 matrix equivalent to a variance-covariance matrix of $\boldsymbol{r}_i(\boldsymbol{\xi})$, and w_i is the weight of i-th pixel. The setting method of $\boldsymbol{\Sigma}_{\mathrm{r}}$ and w_i is explained in Section 3.3. In order to exclude less important pixels and to reduce calculation cost, we define a set $\mathcal{S} \subseteq \{1, ..., n\}$ as the set of index which corresponds to pixels used for calculation of $E(\boldsymbol{\xi})$. Indices i whose magnitudes of gradients $(\frac{\partial \boldsymbol{I}_1(u_i,v_i)}{\partial u}, \frac{\partial \boldsymbol{I}_1(u_i,v_i)}{\partial v})$ are large are included in $\mathcal{S}$, which are supposed to be characteristic region in the image. In the proposed method, the norm of gradient is evaluated at all pixels and the pixels whose norm of gradient are larger than the average are included in $\mathcal{S}$.

Minimization of (3) is a non-linear least squares problem, which takes a lot of time to solve. Therefore, (3) is linearized by the first order Taylor approximation on the assumption that the correction amount $\boldsymbol{\xi}$ is small, and minimization of (3) is reduced to linear least squares problem:

$$\begin{aligned} E(\boldsymbol{\xi}) \simeq\ & \textstyle\sum_{i\in\mathcal{S}} w_i(\boldsymbol{r}_i(\boldsymbol{0}) + \boldsymbol{J}_i\boldsymbol{\xi})^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{r}}^{-1}(\boldsymbol{r}_i(\boldsymbol{0}) + \boldsymbol{J}_i\boldsymbol{\xi}) \\ =\ & (\boldsymbol{R}_0 + \boldsymbol{J}\boldsymbol{\xi})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{R}_0 + \boldsymbol{J}\boldsymbol{\xi}). \end{aligned} \tag{4}$$

$\boldsymbol{J}_i$ is a 3×6 Jacobian matrix of the i-th residual, and $\boldsymbol{R}_0$ and $\boldsymbol{J}$ are stacked matrices of all $\boldsymbol{r}_i(\boldsymbol{0})$ and $\boldsymbol{J}_i$ for $i \in \mathcal{S}$. W is a sparse matrix which consists of diagonally placed $w_i\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}$ and 0 for other elements.

3.3. Minimization of Error Function Robust against Outliers

Minimization of the error function is reduced to the linear least squares problem in section 3.2. It is easy to get the analytic solution of minimization of (4):

$$\boldsymbol{\xi} = (\boldsymbol{J}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{J})^{-1}(-\boldsymbol{J}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{R}_0). \tag{5}$$

However, linear least squares is open to the effect of outliers. If the Image 1 and 2 have few shared field of view, there are many outliers in residuals and registration may fail. Therefore, we employ the Iteratively Reweighted Least Squares (IRLS) method [16], which is robust against outliers and requires small calculation cost. IRLS repeatedly adjusts the weight of each term and solves the least squares, so that the effect of outliers becomes smaller and smaller, without detecting outliers explicitly. Since there are many outliers in the situation of our research, where a pair of images is captured from distant viewpoints and there may be few shared field of view, we propose a new framework of IRLS which is robuster against outliers than the original one.

The proposed flow of the process is illustrated in Fig. 3. First, all weights are set to 1 and the initial solution of $\boldsymbol{\xi}$ is calculated. The weights are then updated by a weight function, using the resulting $\boldsymbol{r}_i(\boldsymbol{\xi})$ and the "weighted" variance-covariance matrix $\boldsymbol{\Sigma}_{\mathrm{r}}$. Minimization of the error function and updating weights are repeated until convergence.

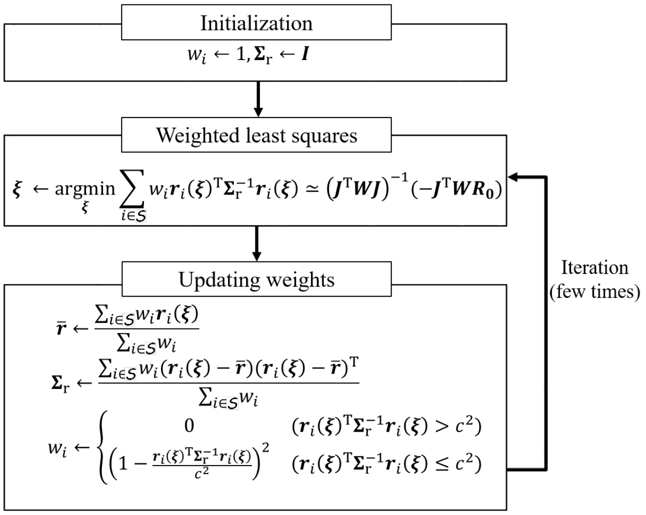

Initialization
$w_i \leftarrow 1, \boldsymbol{\Sigma}_{\mathrm{r}} \leftarrow \boldsymbol{I}$
Weighted least squares
$\boldsymbol{\xi} \leftarrow \underset{\boldsymbol{\xi}}{\operatorname{argmin}} \sum_{i\in\mathcal{S}} w_i \boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi}) \simeq (\boldsymbol{J}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{J})^{-1}(-\boldsymbol{J}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{R}_0)$
Updating weights
$\bar{\boldsymbol{r}} \leftarrow \frac{\sum_{i\in\mathcal{S}} w_i \boldsymbol{r}_i(\boldsymbol{\xi})}{\sum_{i\in\mathcal{S}} w_i}$
$\boldsymbol{\Sigma}_{\mathrm{r}} \leftarrow \frac{\sum_{i\in\mathcal{S}} w_i(\boldsymbol{r}_i(\boldsymbol{\xi}) - \bar{\boldsymbol{r}})(\boldsymbol{r}_i(\boldsymbol{\xi}) - \bar{\boldsymbol{r}})^{\mathrm{T}}}{\sum_{i\in\mathcal{S}} w_i}$
$w_i \leftarrow \begin{cases} 0 & (\boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi}) > c^2) \\ \left(1 - \frac{\boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi})}{c^2}\right)^2 & (\boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi}) \leq c^2) \end{cases}$
Iteration
(few times)

Fig. 3: The proposed IRLS flow. First, all weights are set to 1 and the initial solution of $\boldsymbol{\xi}$ is calculated. The weights are then updated using the resulting $\boldsymbol{r}_i(\boldsymbol{\xi})$ and the weighted variance-covariance matrix $\boldsymbol{\Sigma}_{\mathrm{r}}$. Minimization of the error function and updating weights are repeated until convergence.

We employ the Tukey's biweight function [17], which can completely eliminate the effect of obvious outliers. The well-known Tukey's biweight function consists of a scalar residual r_i, standard deviation of r_i and a parameter c which adjusts sensitivity to outliers. However, $\boldsymbol{r}_i(\boldsymbol{\xi})$ is a vector in the proposed method, so we expand it and the weight of the i-th term w_i is calculated as follows:

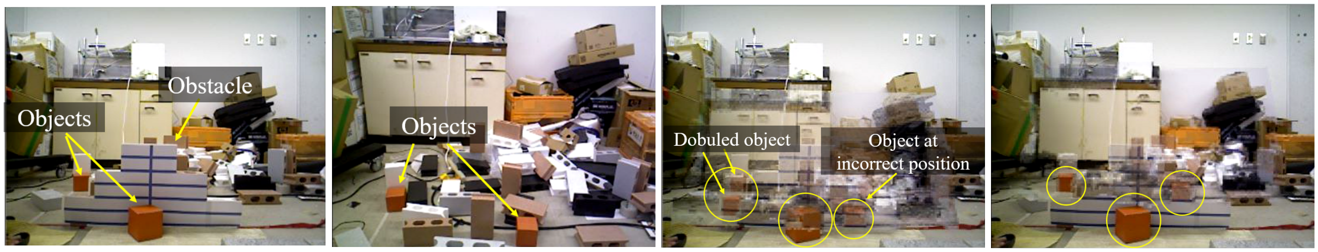
$$w_i = \begin{cases} 0 & (\boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi}) > c^2) \\ (1 - \frac{\boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi})}{c^2})^2 & (\boldsymbol{r}_i(\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathrm{r}}^{-1}\boldsymbol{r}_i(\boldsymbol{\xi}) \leq c^2) \end{cases}. \tag{6}$$

$\boldsymbol{\Sigma}_{\mathrm{r}}$ is a 3×3 matrix equivalent to a variance-covariance matrix of $\boldsymbol{r}_i(\boldsymbol{\xi})$, already appeared in (3). In order to remove the effect of outliers of $\boldsymbol{r}_i(\boldsymbol{\xi})$, we set $\boldsymbol{\Sigma}_{\mathrm{r}}$ as a "weighted" variance-covariance matrix, calculated using weights w_i and the weighted average of $\boldsymbol{r}_i(\boldsymbol{\xi})$, $\bar{\boldsymbol{r}}$.

$$\bar{\boldsymbol{r}} = \frac{\sum_{i\in\mathcal{S}} w_i \boldsymbol{r}_i(\boldsymbol{\xi})}{\sum_{i\in\mathcal{S}} w_i}, \tag{7}$$

$$\boldsymbol{\Sigma}_{\mathrm{r}} = \frac{\sum_{i\in\mathcal{S}} w_i(\boldsymbol{r}_i(\boldsymbol{\xi}) - \bar{\boldsymbol{r}})(\boldsymbol{r}_i(\boldsymbol{\xi}) - \bar{\boldsymbol{r}})^{\mathrm{T}}}{\sum_{i\in\mathcal{S}} w_i}. \tag{8}$$

The minimization of (3) by (5), updating $\boldsymbol{\Sigma}_{\mathrm{r}}$ by (7) and (8), and updating weights by (6) are repeated until convergence. These variables converge in few iterations, so the iterations do not require additional heavy calculation. By using



(a) Input 1.

(b) Input 2.

(c) Before registration.

(d) After registration.

Fig. 4: The experimental result. (a) Input image from the Sensor 1. (b) Input image from the Sensor 2. (c) See through image without registration. Some objects are doubled or appear at incorrect positions. (d) See through image after the registration of the proposed method. The errors of the positional relation of the sensors are corrected and it is easier to confirm the shape or the positions of objects.

the weighted variance-covariance matrix $\mathbf{\Sigma}_{\mathrm{r}}$, effect of outliers on $\mathbf{\Sigma}_{\mathrm{r}}$ is diminished and the weights w_i of outliers are further diminished in (6), which accelerates convergence.

As the linearization is only valid for small $\boldsymbol{\xi}$, we build an image pyramid where the image resolution is halved at each level, and the solution at lower resolution is used as an initialization for the next level. In this way, even large rotation and translation can be handled.

4. EXPERIMENTS AND EVALUATION

4.1. Environmental Settings

We calibrated the positional relation of two RGB-D sensors (ASUS: Xtion Pro Live) first, then moved the Sensor 2 to confirm whether registration of images works even if the positional relation of the sensors has errors. In the experimental environment, there were an obstacle in front of the Sensor 1 and three cubes with a side length of 85 mm which were the objects to be observed. Examples of input images captured from the Sensor 1 and 2 are shown in Fig. 4 (a) and Fig. 4 (b). In Fig. 4 (a) and Fig. 4 (b), two of three objects appear each, while it is impossible to see all three objects in a single image.

The parameter of the Tukey's biweight function was set to $c = 5$, and the number of iteration of IRLS was fixed to 3. These parameters were determined heuristically.

4.2. Registration Results

The see-through image created using Fig. 4 (a) and Fig. 4 (b) without registration is shown in Fig. 4 (c). Although all three objects can be seen in Fig. 4 (c), some objects are doubled or appear at incorrect positions, because of the errors of the positional relation of the sensors.

An example of the result after registration is shown in Fig. 4 (d). After the registration of the proposed method, the errors of the positional relation of the sensors were corrected and it was easier to confirm the positions and shapes of objects.

Table 1: Results of the quantitative evaluation.

Response time	0.5 s
Alignment Precision	5 pixel
Max. of handleable gap	40 pixel
Frame rate	19.4 fps

We evaluated the performance of the proposed method quantitatively, as shown in Table 1. It took about 0.5 s, or 10 frames, for the registration to converge and the position gap was reduced to 5 pixel. It is equivalent to 30 mm at a distance of 1,000 mm from the sensor. The proposed method was able to handle position gap of up to 40 pixel. Using a 2.70 GHz Intel Core i7-6820HQ CPU, the frame rate of the output image was 19.4 fps on average, with a standard deviation of 2.0 fps. Since it is known that the duration of a human moment amounts to 1/18 of a second and that humans cannot perceive what happens within 1/18 of a second [18], this processing speed is supposed to be acceptable for remote operation of robots; the system worked in real-time.

5. CONCLUSIONS

We proposed a dense method to correct positional relation of two RGB-D sensors (6 degree of freedom) to minimize the gap in see-through images. In order to reduce calculation cost, pixels of less importance are efficiently excluded. In order to precisely align images captured from distant viewpoints, the error function is minimized by new IRLS framework which is robust against outliers. The experiments on two RGB-D sensors demonstrated the registration ability in real-time.

It is expected that we can construct a see-though system without external information like joint angle information of a robot arm. This method has wide applications for integration of images captured from multiple viewpoints, especially under a situation where correct positional relation of sensors is difficult to get: multiple mobile robot systems for example.

6. REFERENCES

[1] Masaharu Moteki, Kenichi Fujino, Takashi Ohtsuki, and Tsuyoshi Hashimoto, "Research on visual point of operator in remote control of construction machinery," in *Proceedings of the 28th International Symposium on Automation and Robotics in Construction*. International Association for Automation and Robotics in Construction, 2010, pp. 532–537.

[2] Akihiko Nishiyama, Masaharu Moteki, Kenichi Fujino, and Takeshi Hashimoto, "Research on the comparison of operator viewpoints between manned and remote control operation in unmanned construction systems," in *Proceedings of the 30th International Symposium on Automation and Robotics in Construction*. International Association for Automation and Robotics in Construction, 2013, pp. 772–780.

[3] Tatsuya Kittaka, Hiromitsu Fujii, Atsushi Yamashita, and Hajime Asama, "Creating see-through image using two RGB-D sensors for remote control robot," in *Proceedings of the 11th France-Japan congress on Mechatronics / the 9th Europe-Asia congress on Mechatronics / the 17th International Conference on Research and Education in Mechatronics*, 2016, pp. 86–91.

[4] Akihito Enomoto and Hideo Saito, "Diminished reality using multiple handheld cameras," in *Proceedings of the 8th Asian Conference on Computer Vision*, 2007, vol. 7, pp. 130–135.

[5] Peter Barnum, Yaser Sheikh, Ankur Datta, and Takeo Kanade, "Dynamic seethroughs: Synthesizing hidden views of moving objects," in *Proceedings of the International Symposium on Mixed and Augmented Reality*. IEEE, 2009, pp. 111–114.

[6] Francesco Cosco, Carlos Garre, Fabio Bruno, Maurizio Muzzupappa, and Miguel A. Otaduy, "Augmented touch without visual obtrusion," in *Proceedings of the International Symposium on Mixed and Augmented Reality*. IEEE, 2009, pp. 99–102.

[7] Arturo Flores and Serge Belongie, "Removing pedestrians from google street view images," in *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2010, pp. 53–58.

[8] Songkran Jarusirisawad, Takahide Hosokawa, and Hideo Saito, "Diminished reality using plane-sweep algorithm with weakly-calibrated cameras," *Progress in Informatics*, , no. 7, pp. 11–20, 2010.

[9] Kazuya Sugimoto, Hiromitsu Fujii, Atsushi Yamashita, and Hajime Asama, "Half diminished reality image using three RGB-D sensors for remote control robots," in *Proceedings of the 12th International Symposium on Safety, Security, and Rescue Robotics*. IEEE, 2014, number 43, pp. 1–6.

[10] Christian Kerl, Jurgen Sturm, and Daniel Cremers, "Robust odometry estimation for RGB-D cameras," in *Proceesings of the International Conference on Robotics and Automation*. IEEE, 2013, pp. 3748–3754.

[11] Christian Kerl, Jurgen Sturm, and Daniel Cremers, "Dense visual slam for RGB-D cameras," in *Proceedings of the International Conference on Intelligent Robots and Systems*. IEEE/RSJ, 2013, pp. 2100–2106.

[12] Changhyeon Kim, Sangil Lee, Pyojin Kim, , and H. Jin Kim, "Time-efficient dense visual 12-DOF state estimator using RGB-D camera," in *Proceedings of the 14th International Conference on Ubiquitous Robots and Ambient Intelligence*, 2017, pp. 130–135.

[13] Georg Klein and David Murray, "Parallel tracking and mapping for small AR workspaces," in *Proceedings of the International Symposium on Mixed and Augmented Reality*. IEEE, 2007, pp. 225–234.

[14] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, and David Kim, "KinectFusion: Real-time dense surface mapping and tracking," in *Proceedings of the International Symposium on Mixed and Augmented Reality*. IEEE, 2011, pp. 127–136.

[15] Richard A. Newcombe, Steven J. Lovegrove, and Andrew J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proceedings of the International Conference on Computer Vision*. IEEE, 2011, pp. 2320–2327.

[16] Paul W. Holland and Roy E. Welsch, "Robust regression using iteratively reweighted least-squares," *Communications in Statistics-theory and Methods*, , no. 9, pp. 813–827, 1977.

[17] Albert E. Beaton and John W. Tukey, "The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data," *Technometrics*, vol. 16, no. 2, pp. 147–185, 1974.

[18] Jakob Von Uexkull, "A stroll through the worlds of animals and men: A picture book of invisible worlds," *Semiotica*, vol. 89, no. 4, pp. 319–391, 1992.