

Dynamic 3-D Reconstruction of Human Targets via an Omni-Directional Thermal Sensor

Emrah Benli, *Member, IEEE*, Jonas Rahlf, and Yuichi Motai¹, *Senior Member, IEEE*

Abstract—We explore the dynamic 3-D reconstruction (D3DR) of the target view in real-time images from the omni-directional (O-D) thermal sensor for intelligent perception of robotic systems. Recent O-D 3-D reconstruction methodologies are mainly focused on O-D visible-band vision for localization, mapping, calibration, and tracking, but there is no significant research for thermal O-D. The 3-D reconstruction from O-D images and the use of O-D thermal vision have not been sufficiently addressed. The thermal O-D images do not provide sharp-edge boundaries as in color vision cameras due to texture and mirror distortion. In order to fully address O-D thermal 3-D reconstruction, we proposed the D3DR method that dynamically detects the target region and densely reconstructs the detected target region to solve the non-sharp-edge boundaries' issue. We analyzed several different imaging positions, different baseline distances, and target distances with respect to the robot position for the best coverage of the target view with a minimum reconstruction error. We also look at the optimum number of observations for reconstruction using an optimization to find the compromise between accuracy, methodology, and number of observations. The benefits of this method are the accurate distance of the target from the camera, high accuracy, and low computation time of 3-D reconstruction.

Index Terms—Far infrared sensor, human-robot system, mobile robot, omni-directional sensor, sensor application, thermal vision, 3D reconstruction.

I. INTRODUCTION

OMNI-DIRECTIONAL (O-D) vision systems in recent years have been key elements of many applications such as robotics, automotive, home automation, security, and mobile systems. These camera systems offer the advantage to view a wide angle of environments of mobile systems and to assist in tracking targets [1], [2]. O-D cameras decrease the required number of sensors in comparison with standard camera systems [3]–[5] since O-D covers a much wider field of view. Most studies, however [1], [6]–[9], are based on “visible-band” O-D systems. Those systems do not allow light independent tracking and 3D reconstruction for critical

Manuscript received May 24, 2018; accepted June 26, 2018. Date of publication July 13, 2018; date of current version August 10, 2018. This work was supported in part by the U.S. Navy, Naval Surface Warfare Center Dahlgren, in part by the Ministry of National Education of Turkey, and in part by DAAD Rise Worldwide. The associate editor coordinating the review of this paper and approving it for publication was Prof. Tarikul Islam. (Corresponding author: Yuichi Motai.)

E. Benli is with the U.S. Army Research Laboratory, Adelphi, MD 20783 USA (e-mail: benlie@vcu.edu).

J. Rahlf is with the Department of Production Engineering, Bremen University, 28359 Bremen, Germany (e-mail: j.rahlf@uni-bremen.de).

Y. Motai is with the Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA 23284 USA (e-mail: ymotai@vcu.edu).

Digital Object Identifier 10.1109/JSEN.2018.2856106

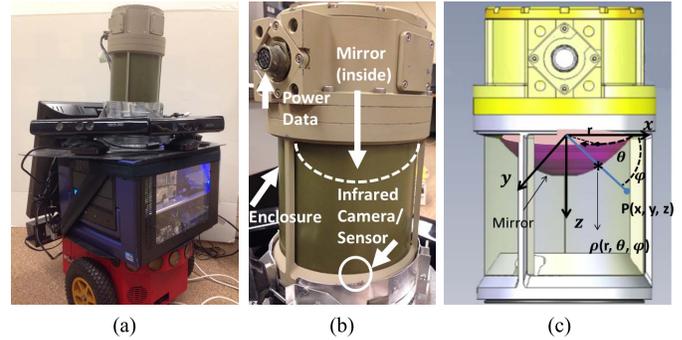


Fig. 1. The O-D Infrared Camera (a) the mobile robot equipped with O-D IR camera, (b) the out looking image of the O-D camera with the placement map, and (c) representative inner structure of the O-D camera with the mirror coordinate system. Symbolic reflections from an object point $P(x, y, z)$ and its reflection coordinates on the mirror are given as $\rho(r, \theta, \phi)$.

missions such as military and rescue operations. Since the main interest in such missions is human, obtaining additional information about a human target for longer time operations is critical [10], [11]. O-D infrared cameras help to obtain more human-target features since IR images provide a clear pattern for human body temperature with longer time. O-D view helps to track any orientation of the target around the robot. One drawback of infrared sensors is that infrared O-D has lower resolution than visible-band camera sensors [12]. This problem makes improving the visibility of a human target from the thermal images significant. 3D reconstruction of thermal view is an efficient way to improve visibility and provides an enhanced view of a human target. 3D reconstruction is refining the visibility and maximizing obtained information [13]–[15], so that shape and distance of the target is being done by stereo cameras. A stereo O-D IR imaging could be considered an efficient way for 3D reconstruction to develop the target scene; however, another problem is occurring with bigger size of the thermal sensors with respect to visible-band sensors. In this study, we use O-D infrared images from the O-D thermal sensor mounted on the robot so that multiple images in the O-D thermal sensor can capture additional spatial information for 3D reconstruction of the robot’s environment and its static target.

A single O-D infrared sensor, mounted on top of the mobile robot, as shown in Fig. 1(a), obtains the infrared images for 3D reconstruction while the robot is in motion. Fig. 1(b) displays the O-D camera used in our application and the configuration of its parts. Fig. 1(c) shows the coordinate system of the camera, and the reflection of an object point on the parabolic mirror and its representative coordinates. The parabolic mirror

reflects rays from objects towards the camera sensor which is placed on the mirror's z -coordinate axis.

Many different approaches have been proposed on O-D cameras for 3D reconstruction in previous articles. Using different mirrors with distinct orientations of the camera is proposed in [1], [2], [6], [16], and [17]. Several cameras are utilized to reconstruct, stereo case, in [2] and [9]. Another way to reconstruct a 3D view of O-D images is by using several images from one O-D sensor [18]–[21]. However, reconstruction of O-D “infrared” images is rare in this area of study.

The academic contribution of this study, called Dynamic 3D Reconstruction (D3DR), is to propose the O-D 3D reconstruction approach using a thermal camera with respect to current methods, which are limited to O-D visible-band sensors. The thermal images do not produce sharp edges as visible-band cameras would have for textures of equivalent temperature for background objects in source images. Passive objects such as desks and chairs do not provide any contrast in the detection of thermal boundaries while human-like objects provide contrast. Far regions also contain noisy feature points which are not consistent in consecutive O-D images throughout the robot's trajectory. The parabolic mirror structure of the O-D camera causes these noisy feature points. Parabolic coordinates have distortion mostly in the outer and inner region of the O-D images. These regions correspond to the objects that are far or that are very close in distance from the robot. As a result, the reconstructed points do not represent the real object points accurately. This affects both the distance and shape of the objects being tracked by the mobile robot. *The newly proposed method will address this by using a combination of two algorithms correlation based target detection and dynamically orientated feature selection that will focus on a target region in order to increase the number of feature points around the objects of interest instead of the noisy regions.* Since the target position on the image is moving, the target region in the robot's view is being dynamically searched and detected as a 3D reconstruction region. We used four datasets with different object orientations and environments for the experiments. To find more accurate results, we selected the best number of observations and chose baseline distances between these points.

This paper is organized as follows: Section II discusses previous related works on O-D cameras and 3D reconstruction of O-D images. Section III describes the geometrical mathematics for the O-D camera and 3D reconstruction from these cameras. In Section IV, we proposed our method on 3D reconstruction of O-D infrared images captured from the robot. Section V presents experimental results of our method. Finally, in Section VI, the conclusion and future work is presented.

II. RELATED WORKS

We cover relevant studies on 3D reconstruction of target scenes using a 360 degree thermal imager on a mobile robot. The following subsections will first describe Section II.A O-D cameras based on their methods and mechanical designs to reconstruct images in 3D both for color and infrared cameras. Then, in Section II.B covers 3D reconstruction from O-D

TABLE I
O-D SENSORY IMAGING APPROACHES

Sensor	Operation Requirements (Light Conditions)	Sensor or Mirror Requirement	Resolution
Color O-D single mirror [22], [26]	High	1	High
Color O-D multiple mirrors [1]–[2], [6], [16]–[17]	High	>2 Mirrors	High
Stereo O-D [2], [9], [23], [24], [31]	High	>2 Sensors	High
IR O-D [25], [32]	Low or High	1	Low

cameras to obtain better results in terms of 3D reconstructed robot view that returns improved results compared to previous methods [1], [2], [9], [18], [19], [22]–[24].

A. Omni-Directional Camera

Among the existing O-D cameras, for 360-degree reconstruction, several approaches exist: 1) a mirror [22], [25]–[30], 2) multiple mirrors, stereo case by mirrors, [1], [6], [16], [17] 3) one mirror with perspective camera combinations or multiple camera sensors in a hemisphere pattern, [2], [9], [23], [31], 4) and the O-D imaging from infrared cameras with the same approach, [25], [32]. Table I shows the details of four different O-D sensory settings: their lighting requirements for sufficiently low noise, required number of camera sensors, and resolution of images from the sensors. There exists several approaches for the use of mirrors. The image is taken from a parabolic or similar type of mirrors via a camera offers advantages to better analysis of the environment.

The first approach, using a mirror with a traditional color camera to reconstruct the O-D images, is one of the most broadly used methods [22], [25], [26], [28], [29]. A color camera captures the image from a catadioptric mirror fixed across from camera. A spherical camera model and its calibration methods are explained in [16]. A new O-D mirror design can also take panoramic images as proposed in [6].

The second approach obtains a 3D view of the monocular image using multiple mirrors. The method [1] describes a 3D reconstruction via four parabolic mirrors. Another method [2] uses two mirrors and a camera to capture two images from these mirrors. These mirrors, the inner part and outer part of the single camera, provide two images to be reconstructed. In order to get a good 3D reconstruction result, stabilization is a key factor as motion can easily affect the 3D reconstruction process. The trajectory estimation problem is discussed in [17] as well.

The third approach is to reconstruct the 3D view using stereo camera sensors [9], [23], [24] similar to the stereo perspective cameras. Methods [9], [23] use a perspective camera in addition to an O-D camera. The geometric constraints for the placement of these cameras are discussed [9] in a previous work. A hybrid method is an effective way to match the feature points in the hybrid images from an O-D camera and perspective camera automatically [23]. Images from O-D cameras have a wide vision range with low resolution, so the perspective camera helps to enhance the 3D reconstruction. The variations of using two O-D cameras in vertical and

horizontal configurations represent another O-D stereo vision setup [24]. A method [31] is also proposed the multiple sensors in a hemisphere pattern to obtain the 3D scene.

The fourth approach includes infrared cameras to obtain a 360 degree O-D image that are the same as the methods as the previous two methodologies [25], [32]. The difference applied in this method is the replacement of the color camera and mirror material in the O-D camera system by an infrared camera. Thus, the O-D infrared cameras use the same mirror structure to attain the infrared images. However, using an infrared camera has certain physical limitations to provide a good resolution and information to track objects. We adopt the O-D camera setup with one mirror and infrared camera to explore [1], [33]–[37] in an intense 3D information manner as shown in Section IV.

B. 3D Reconstruction From Omni-Directional Camera

3D reconstruction from infrared O-D cameras is feasible using similar methods as the O-D color camera approach by utilizing mirrors [1], [2], [6], [16], [17], or using several cameras, and stereo methods [9], [23]. The 3D reconstruction is also possible by the means of one O-D camera and numerous images along with camera and robot movement [8], [18], [29]. In this method, the camera captures several images at different positions. This method may be considered the same as the stereo method or using several mirrors instead with only one camera. The accuracy of 3D reconstruction for O-D images is obtained from an image sequence that is better than using two or less images such as a single view reconstruction [19], [29], [34]. Unlike the predefined position of the cameras in a stereo setup [2], [9], [38], method [21] uses the robot's odometer.

Studies about infrared O-D cameras are rare [25], and resolution is low in infrared images. We propose instead a 3D reconstruction approach to acquire a high density 3D reconstructed view from infrared images. The robot captures images through an O-D camera from multiple positions to reconstruct the 3D scene by the proposed algorithm. Improving the density of 3D reconstructed feature points assists in tracking objects as opposed to tracking in 2D infrared images.

III. GEOMETRY FOR 3D RECONSTRUCTION

Section III gives the mathematical fundamentals of 3D reconstruction in the basic foundation. First, we examine the geometry of the O-D camera by obtaining the projection via a parabolic mirror in Section III.A, and we then formulate a 3D reconstruction by employing the O-D camera in Section III.B.

A. Geometry of Omni-Directional Camera

This section explains the geometry of an O-D camera and the reflection of a projected point in space from a parabolic mirror. Fig. 2 shows the projections and parabolic mirrors [7], [22]. Equation (1) derives the coordinates of the real points, e.g. $P^W = (x \ y \ z)$ in Fig. 2, on the O-D camera mirror;

$$P_m = (x_m \ y_m \ z_m) = \lambda [R_m, -R_m t_m] P^W \quad (1)$$

where R_m, t_m is the transformation from world to the mirror.

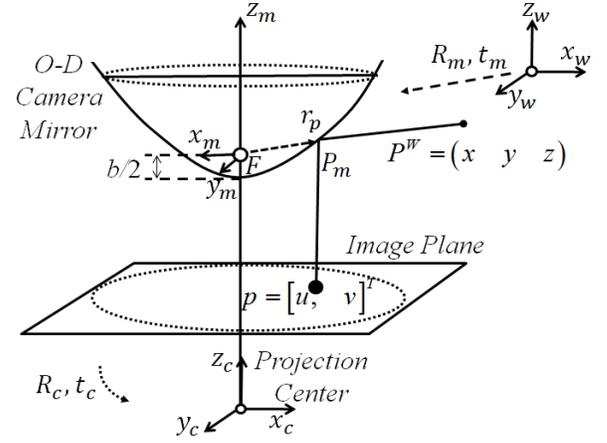


Fig. 2. Projections of the parabolic mirror. Simplification of 3D to 2D geometry is done by using r representing x and y axes, and symmetry about z axis.

$P^W = (x \ y \ z)$ is a 3D real point in space and $p = [u \ v]^T$ is the projection of said 3D point from the parabolic mirror onto an image. The point's reflection on the parabolic mirror has a ray vector of $r_m = [x_m \ y_m \ z_m]$ and this ray vector is converted into another point $p = [u, v]^T$ on the image. The properties of our camera's mirror meets the single viewpoint condition by using parabolic mirror properties. The properties for a parabolic O-D sensor model provide a ray vector from the mirror center point to the real world point [30]. Equation (2) provides the image pixel point $p = [u, v]^T$ by using the real world point $P^W = (x \ y \ z)$.

$$p = K R_c \begin{bmatrix} \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} [R_m, -R_m t_m] \\ 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (2)$$

where parameter K , the intrinsic matrix, calculated by the camera calibration process, and R_c is the camera sensor rotation. The real world coordinates are $(x \ y \ z)$ and equation (3) calculates the scale λ .

$$\lambda^2 (-x^2 - y^2) + \lambda (2bz) + b^2 = 0 \quad (3)$$

where $b/2$ is the distance between the focal point F of the parabolic mirror and the vertex of the mirror.

The point $P_m = (x_m \ y_m \ z_m)$ which lies on the mirror surface can be obtained by using known image point $p = [u, v]^T$ from (4).

$$[x_m \ y_m \ 1]^T = R_c^T K^{-1} p \quad (4)$$

We can find the third coordinate from $z_m = ((x_m^2 + y_m^2)/2b) - (b/2)$, and the mirror point $P_m = (x_m \ y_m \ z_m)$, [30]. The coordinates of mirror points provide the ray vector $r_m = [x_m \ y_m \ z_m]$ of the real world point from the focal point of the camera.

We illustrate the mechanical architecture and view diagram of the O-D IR camera and the geometric principle of the camera in Fig. 3. The effective vertical view range is 55 degrees in total, 5 degrees upper view and 50 degrees lower view.

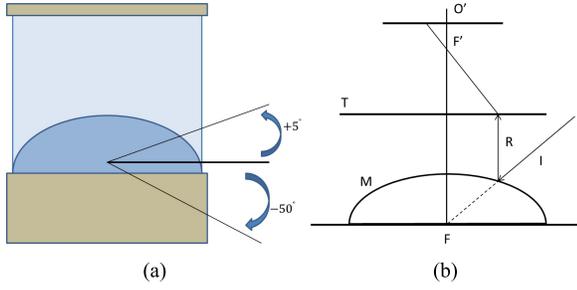


Fig. 3. View diagram and geometric principle of the O-D camera. (a) The elevation angles. (b) The reflection of the ray in the O-D camera.

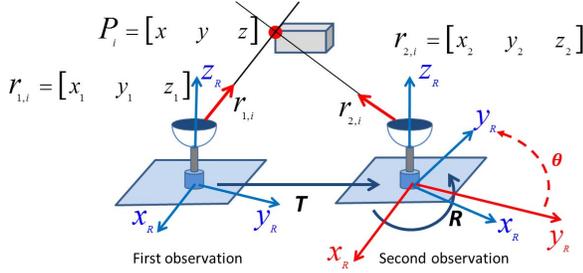


Fig. 4. 3D reconstruction from two observations.

B. 3D Reconstruction From an Omni-Directional Camera

3D reconstruction of the O-D image is formulated by using $r_m = [x_m \ y_m \ z_m]$ ray vectors of a feature point from different images. This ray vector, given in section III A, displays a reflection point of a ray on the mirror from the projection center [18]–[20], [30]. The method selects the points by focusing on the target region, which gives reliable tracking of feature points and precise calculations compared to previous methods. First, the essential matrix E is the required factor which satisfies;

$$r_2^T E r_1 = 0 \quad (5)$$

where $r_{1,i} = [x_1 \ y_1 \ z_1]$ and $r_{2,i} = [x_2 \ y_2 \ z_2]$ are the ray vectors of corresponding points from two images. The essential matrix offers information about the orientation and positioning differences between these observations. The matrix E is the key element for calculations, since the 3D object coordinates are measured by the changes of the camera position [39]. The singular value decomposition from the essential matrix E is proposed in [18] and [19] that finds the translation vector T and rotation matrix R .

3D coordinates of real points are obtained by triangulation of the position of observations. To find the 3D coordinates of feature point P_i , we used the rotation matrix R and translation vector T , the ray vectors $r_{1,i}$, and $r_{2,i}$, as shown in Fig. 4.

IV. TARGET ORIENTED DYNAMIC 3D RECONSTRUCTION (D3DR) FROM A FAR INFRARED OMNI-DIRECTIONAL CAMERA

A new method (D3DR) for 3D reconstruction of images from infrared O-D cameras is implemented with target region detection. The reconstruction region is dynamically searched in an image sequence while the robot moves. The low resolution of infrared images makes the requirement of 3D reconstruction significant for infrared images to get higher density for 3D reconstructed robot view. Section IV.A shows

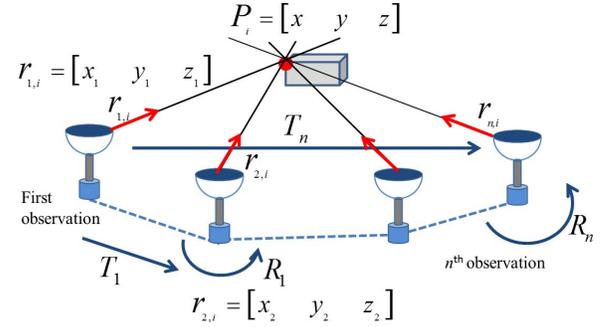


Fig. 5. n observations for the calculation of 3D coordinates. $P_1 = [x_1 \ y_1 \ z_1]$ is corresponding feature point coordinate from the first observation position while $P_2 = [x_2 \ y_2 \ z_2]$ is the same feature point on the second image from a second observation.

the reconstruction of images from two observations with triangulation, and multiple observations while updating the 3D coordinates with bundle adjustment as in Section IV.B. Then, Section IV.C explains the target orientated reconstruction with the sensor position by using the feature points. Finally, we determine the optimal numbers of observations in Section IV.D.

A. Triangulation From Omni-Directional Image Pairs

A high number of observations increases the density of the 3D reconstructed robot view. Fig. 5 illustrates the calculation of 3D coordinates of real points from two observations. Thus, the improved information of low resolution infrared images helps to improve tracking the objects. The previous work uses n observations to calculate the essential matrix. We use this matrix to find the translation and the rotation information for the camera position. The purpose of using the essential matrix E is obtaining the precise position of n observations. After the position analysis, we apply a triangulation as in a stereo setup.

In our proposed method, we use each consecutive image pair to calculate the 3D coordinate of the i^{th} feature point P_i by triangulation until the last observation. The baseline distance between the observations and the ray vectors, $r_{1,i}$ and $r_{2,i}$, of the feature points are used for triangulation, Fig. 4. The feature point $P_i = [x \ y \ z]$ and its coordinates from the first and second observation are points $P_1 = [x_1 \ y_1 \ z_1]$ and $P_2 = [x_2 \ y_2 \ z_2]$.

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \left(\begin{bmatrix} P_1 \\ -R P_2 \end{bmatrix} [P_1 \ -R P_2] \right)^{-1} \begin{bmatrix} P_1 \\ -R P_2 \end{bmatrix} T \quad (6)$$

R , the rotation matrix and T , the translation vector provide the relationship between coordinates of the feature points from the first and the second observations. The solution of (6) gives the relationship between (z_1, z_2) for depth information of the feature point from two different perspectives. The world coordinates of the feature point are obtained by using the first observation coordinates as a reference and the results of the depth calculations.

B. Multiple Observations for Omni-Directional Images

For higher density of a 3D robotic view, we perform the reconstruction with two or more different sensory settings, Fig 5. The previously proposed methods [18]–[20] uses these

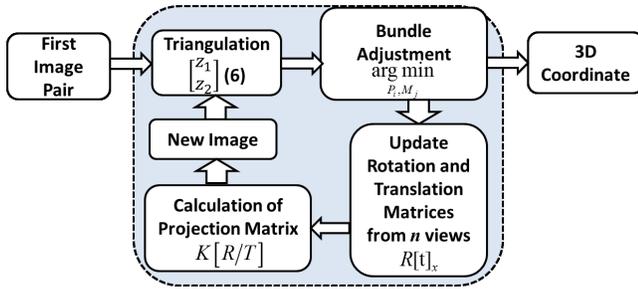


Fig. 6. Flowchart of updating a 3D coordinate of the feature points.

kinds of techniques. However, those methods take a longer time to estimate the general shape of the robot's environment rather than the precise distance of a target. The images taken from these observations get us a high density 3D view of infrared images rather than using the method with only two images.

Triangulation of image pairs provides 3D coordinates of the first image pair. Then, bundle adjustment finds the optimal coordinates from each extra image, shown in Fig. 6. Another reconstruction method is merging partial reconstruction. This method uses more than two images to find better results for 3D coordinates of the feature points. In order to get more precise results after bundle adjustment, we use the output of the merging partial reconstruction method instead of triangulating image pairs at each iteration.

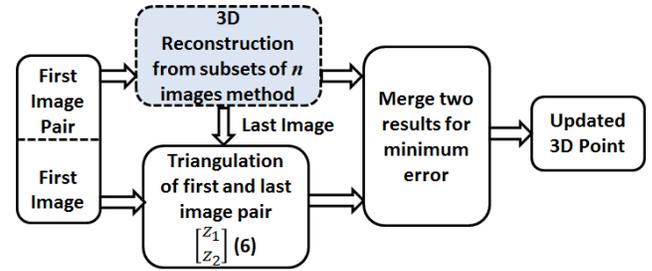
During the reconstruction of the feature points up to the last observation, we have $n-1$ numbers of 3D coordinate information for the same feature point. Since some of these coordinate results are far from the average value, the algorithm removes those irrelevant outlier values. Finally, the candidate of 3D coordinates of each feature point P_i are calculated from consistent results for the calculation from n observations.

After finding the 3D coordinates, we apply bundle adjustment to obtain the optimal 3D structure. Bundle adjustment utilizes observed image points as well as projected 3D points to find the optimal 3D point and projection matrices [18]–[20]. This process is specifically for the minimization of the Euclidean distances between observed and projected points. Each new feature point updates the 3D coordinates through bundle adjustment.

Projected coordinates, $p_{i,j}$ observed points, j^{th} image i^{th} feature point and $\hat{p}_{i,j}$ reprojection of reconstructed 3D points, are derived by multiplication of $M_j = K [R/T]$ and point P_i . Minimum Euclidean distance is calculated using observed, $p_{i,j}$, and reprojected feature points, $\hat{p}_{i,j}$. R is a rotation matrix and $T = [t_x \ t_y \ t_z]^T$ is a translation vector of the camera. The M_j projection matrix and K matrix (7) are given by the calibration matrix including intrinsic camera parameters focal lengths, α_u , α_v aspect ratio s and principle point $p (u_0, v_0, 1)$.

$$K = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

From every n image sequence, we update the rotation matrix, the translation vector, and use them to obtain the projection matrix for each image sequence. The methods


 Fig. 7. Flowchart of merging 3D coordinate from subsets of n images and triangulation of the first and last image by using projection matrix of all images.

previously proposed, [18]–[20], calculate the orientation of the robot in a similar way; however, the error is relatively high to which causes a weak 3D reconstruction and does not work for O-D IR images. If the estimations are not made for the shape of the environment for the precise distance of the target, our method gives an improved result. The algorithm constructs the robot trajectory from the rotation matrix, the translation vector, and removes inconsistent values from the trajectory of robot. This helps to avoid the effects of localization error on reconstruction. Once we have the camera orientation, the first image and last image are sent to the triangulation step to find new 3D coordinates of the feature points. The bundle adjustment step updates the reconstructed 3D coordinates with a projection matrix for every n image sequences. The optimal value of n will be determined by comparing the errors of a 3D calculation from numerous images in section V.F.

After reconstructed coordinates of every n image's subset are updated via bundle adjustment, it is expressed as the highlighted box in Fig. 7 and is shown by Fig. 6. The algorithm uses the first and the last image for triangulation with the projection matrix from all images. A comparison applies these calculated coordinates, the rotation matrix and the translation vector with the values from n image reconstruction and the position matrix. For the final updated coordinates, we merge these two methods to find the optimal point for available feature points for n images. Every image pair in the sequence is used for the reconstruction, then the first and last images of this sequence are used to compare the robot orientation consistency between two positioning results. The algorithm also includes the reconstruction results for the first and last image pair as well. Thus, we obtain the major feature points sensitively.

C. Target-Orientated Reconstruction With Sensor Position

The proposed D3DR consists of two steps. The first step is detecting the target region, and the second step is detecting while tracking the feature points within this region. D3DR reduces the region of interest to get rid of noisy background for the next step. We do this noise reduction by dynamic separation of unreliable corner points from the target region. Green feature points are considered as the noisy feature points because of the O-D mirror, reflection noise of the camera structure in the IR images, and the texture of the objects which have the same temperature. These noisy points, caused by the problem of low resolution and non-sharp edges of objects with

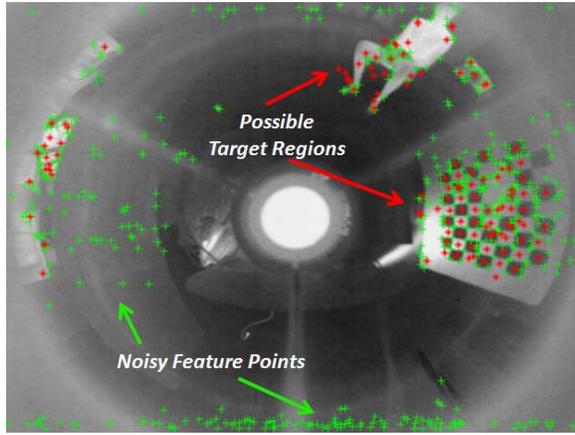


Fig. 8. The candidates of the target region are represented with the red point clouds that have intense feature detection applied on to them. Noisy feature points are not included for 3D reconstruction in the target regions.

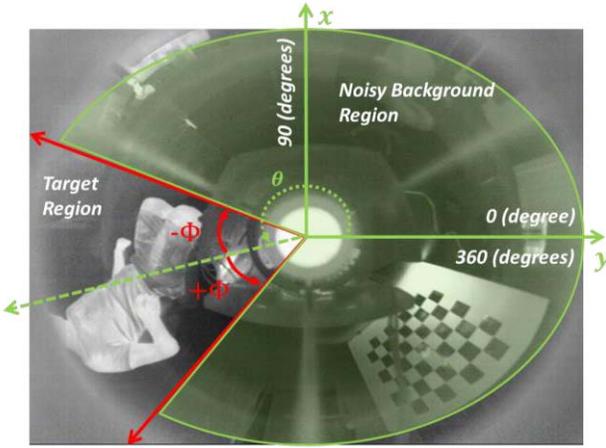


Fig. 9. Reducing interest region to avoid noisy background and achieve better reconstruction of the target.

similar temperature, are shown in Fig. 8. After detection of the target region, the algorithm restricts the O-D IR image to the area with virtual rays for the next feature point detection process shown in Fig. 9. These limitation rays dynamically move with respect to the next target position in the following images. Therefore, target region candidates are detected and considered into the target detection method; based on the correlation between the target templates data base that includes a set of human targets and the original image as the first step. We then use normalized 2D cross-correlation.

The algorithm determines the target regions automatically by using highly correlated regions. The method selects the highest correlation result from each template and analyzes human body characteristics in the candidate target region. The scaling is not used as a criteria to find the highly correlated regions. Instead of scaling the templates, we analyze the correlated regions since the target view was consistent with our data set. A threshold with respect to human body temperature applies to the images and selects accordingly the big blobs by the size of the human target. We also use a ratio of height and width for a possible target to eliminate those blobs that do not correspond to human target region characteristics. Then, we localize the region of interest from the direction of the highly correlated pixels according to a retrieved target size.

The feature detection method uses the eigenvalues of each pixel in the image, which finds the green feature points and more intense corner feature point information. Since noisy background is removed from the interest area of the image at the previous step, the detected points are mostly from the target in the region of interest. The green area in Fig. 9 is assigned as an out of interest area so the focus is on the target region. The algorithm increases the number of feature points in this obtained target region and tracks this region dynamically across the images in the data set. Thus, the noisy green feature points that are not detected before the target detection step stay out of the target area. We then detect the feature points which are in the target region in the second step, known as the feature point detection process. We use the feature detection method from the Matlab computer vision toolbox [40] using the minimum eigenfeature method. The feature points are selected from well localized pixels of minimum eigenfeatures after more intense feature point coordinates are obtained.

Algorithm 1: Target Region Detection Algorithm

For each image I in the image sequence
 $TC \leftarrow$ Target region candidate via Template Correlation (Highest correlation);
 $T \leftarrow$ Compare the human body characteristics in *thresholded image* (Size > 10000, 0.9 > human body temperature > 0.5, target ratio > 1.53);
 $TR \leftarrow$ Find the *Target Region* (center and $\pm\Phi$ range for *Target Region*);
 For each target region (T)
 $TF \leftarrow$ Find the Feature points in Target Region via *feature detection with respect to eigenvalues*;
 $P_i \leftarrow$ Reconstruct Feature Point (TF) via *Triangulation*
 $P'_i \leftarrow$ Update the Reconstructed Point (P_i) via *Multiview Triangulation*

The position of the target on the images updates the coordinate setting of the camera as part of dynamic 3D reconstruction. This process is given in Algorithm 1. If the robot (camera) is moving away from the target, we update the rotation matrix R in order to have a parallel movement to the target for the best reconstruction angle. We then add the rotational angle θ , in Fig. 9, about the z axis to the R rotation matrix by the following equation,

$$R' = \begin{bmatrix} R_{xx} & R_{yx} & R_{zx} \\ R_{xy} & R_{yy} & R_{zy} \\ R_{xz} & R_{yz} & R_{zz} \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The new rotation from R' updates relative coordinate settings for the sensor position. Finally, we calculate the second rotation matrix and translation vector from the first and last images to be used for the second triangulation process to merge the two results. A flowchart of the process is given in Fig. 10.

$$M_j = K [R'/T] = K \begin{bmatrix} R'_{xx} & R'_{yx} & R'_{zx} & t_x \\ R'_{xy} & R'_{yy} & R'_{zy} & t_y \\ R'_{xz} & R'_{yz} & R'_{zz} & t_z \end{bmatrix} \quad (8)$$

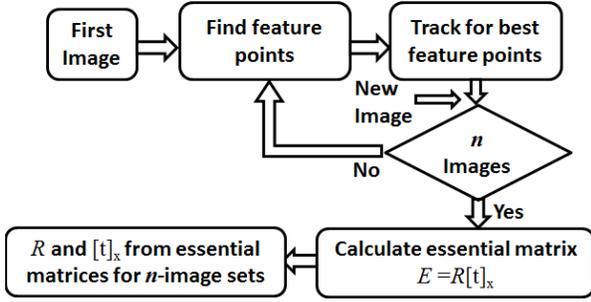


Fig. 10. Flowchart of obtaining essential matrices for each n images sets to calculate rotation and translation matrices.

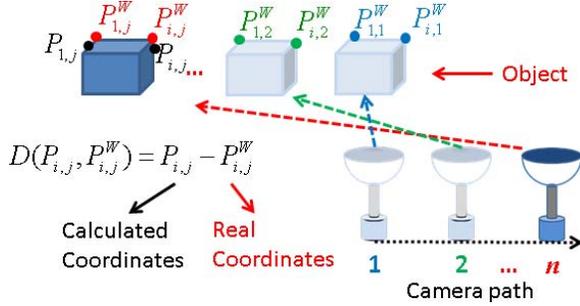


Fig. 11. Feature points $P_{i,j}$ are seen from the first observation in blue, from second observation in green and last observation in red.

The projection matrix M_j can be calculated by (8) where K is an intrinsic parameter matrix and can be derived from the calibration process with focal lengths parameters, α_u , α_v and principle point $p(u_0, v_0, 1)$. Since we have square feature point pixels, we consider aspect ratio s as zero and express K by (9),

$$K = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (9)$$

The current coordinate of the camera is calculated by intrinsic parameters, as well as the rotation and translation matrices. Thus, we can construct the projection matrix to calculate 3D coordinates by triangulating after the update for the projection matrix of every n images.

D. Optimal Number of Observations

The optimal number of observations is selected by several criteria within the reconstruction process such as accuracy of the reconstructed coordinates, and the limitation of the system memory to save images from different observations. Some other studies [1], [2], [22], [23] use such criteria, which we adapted into our O-D camera setting.

First, we calculate the accuracy of reconstructed coordinates from the difference between real coordinates and the retrieved coordinates in Fig. 11. We then measure the real coordinates of the feature point with a tape measure in the experiment environment and use average distance of the human body as approximated real coordinates of each reconstructed point. For this process, we select continuous featured points, then, we track them during the image sequences. The algorithm reconstructs each feature point pixel from the first observation to the end of the sequence. The number of images in a

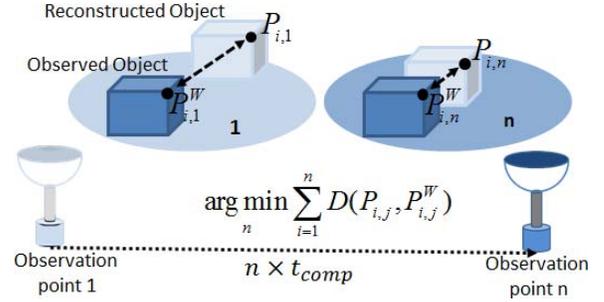


Fig. 12. An illustration of error change from the first triangulation to n^{th} triangulation.

sequence for a feature point depends on the maximum distance between the object and camera in an efficient reconstruction limitations. In Fig. 12, we can see the change of error between the reconstructed points and the real world points as the number of observations increases. At the beginning, the error is high and gets smaller while the number of observations is getting larger. However, when the object is far from the camera then the changes in feature points are not measured precisely, so, reconstruction error increases.

In these image sequences, we sum up the errors of each feature point. We obtain a cost function for the average of total error and number of images at the end of the process. In order to find the optimum number of observations, the algorithm selects the minimum error. This process is applied for different data sets to find the optimized number using (10).

$$\arg \min_n \sum_{j=1}^m \sum_{i=1}^n D(P_{i,j}, P_{i,j}^W) \quad (10)$$

where m is the total number of feature points in images, $P_{i,j}$, that is a retrieved feature point coordinates of real feature point coordinates, $P_{i,j}^W$. The minimum total of differences of these two coordinates, the reconstruction error, provides the optimal number of observations as n .

The required computation time, the second criterion, is t_{comp} , which is required to calculate 3D coordinates for each image pair. The number of images, n , also varies with the baseline distance, Δ_{base} , of the imaging position of the robot, $n = \Delta_x / \Delta_{base}$ where Δ_x is the total displacement of the robot.

As the last criteria for the number of images, we consider RAM consumption for the data set with respect to RAM consumption of the system, RAM_{cons} . If the data size exceeds the RAM consumption of the system, the algorithm determines the number of observations, n , by the RAM limitation of the system. Each image has an average size of 0.156 MB and the total size of the data set is calculated by, $Size_n = n \times 0.156 \text{ MB}$. Three constraints decide the optimal number of images;

$$\begin{aligned} t_{comp} &< \Delta_{base} / V_{Robot}, & 2 \leq n &= \Delta_x / \Delta_{base}, \\ n &\leq RAM_{cons} / 0.156 \end{aligned}$$

For these constraints, we want to find the minimum number of images n , for the minimum value of the cost function, $\arg \min_n = f(error^2, n \times t_{comp}, Size_n)$ which is the sum of three objectives: the RAM consumption ($Size_n$) of data,

reconstruction time ($n \times t_{comp}$), and reconstruction error of the feature points $error = D(P_{i,j}, P_{i,j}^W)$. Since reconstruction error depends on its distance from the real coordinates of the feature points, there is the possibility for it to be negative or positive. For that reason, we calculate the reconstruction error during the calculation of the cost function for each iteration. Reconstruction accuracy is the most important criterion to determine the number of observations.

V. EXPERIMENTS

The experiment section of this paper is organized in the following subsections: First, we discuss the information about the data sets in Section V.A. Second, we compare 3D reconstruction of multiple views for different target objects with respect to the number of feature points of objects, as in Section V.B. Then, Section V.C analyzes the reconstructed points according to distance from the camera. Consequently, we compare the reconstruction with different baselines of observation positions in Section V.D. Then, we organize section V.E as the decision of the optimal number of images in the sequences with respect to accuracy of the 3D coordinates, the memory requirement, and computational time. Finally, in Section V.F, we compare the proposed method along with other methods in terms of sensory settings and reconstruction processes.

A. Hardware and Data Set Information

The hardware of the mobile robot platform consists of three main components. First component is the mobile robot Pioneer 3-DX with an on board PC for fully autonomous intelligent control. The second component is a Windows 7 computer with an Intel i7 processor that runs the image processing with Matlab. Third main component is the O-D IR camera sensor that captures 360 degrees omni-directional thermal images [41]. IR images have a gray level corresponding to the temperature of objects with 480×640 resolution and the imaging frequency of the sensor is 30fps. In Section I, Fig. 1 shows the mobile robot platform with the O-D IR camera sensor and its mounted windows computer. The calibration of the O-D IR camera was done by implementation of the O-D camera calibration toolbox [42]. Since the current toolbox works for only visible-band cameras, we use a heated grid pattern constructed from metal based materials in order to reflect the heat instead of light.

We obtained the data sets for reconstruction from several different environments and object orientations. One of the data sets contained more texture and complicated objects in a room condition. We can see these objects near the human target or around the O-D sensor, Fig. 13(b), (c), and (d). The warm objects were used to represent the outdoor interference such as vehicles with their hot tires, reflection of sun rays or other warm objects. We also obtained some data sets in a small area with a moderate number of surrounding objects. Another data set included images while the robot was moving through a corridor with less texture where objects were easy to detect. And the last data set included the rotated path around the detected main target to extend the trajectory length with a stable distance from the target. The human target was static in all data sets in order to obtain the correct tracking of the

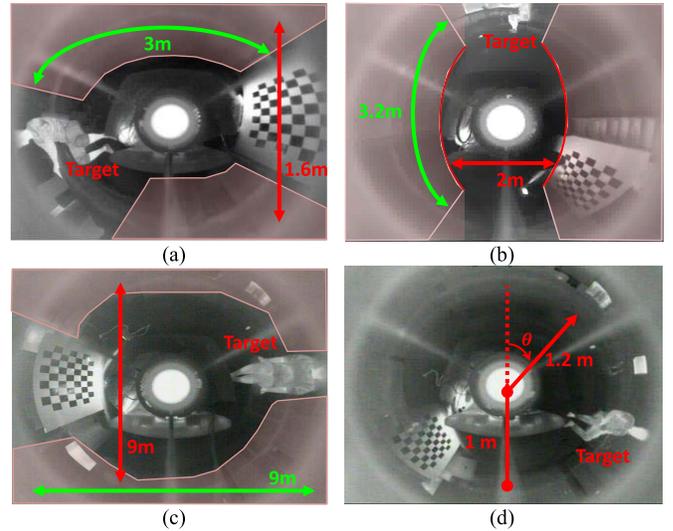


Fig. 13. Representative imaging conditions: (a) long time, (b) long path, (c) large area, and (d) rotation throughout the path in the large area.

TABLE II
INDOOR OBJECTS SCENE

Object Type	Temperature	Contrast	Reconstructability
Human	High	High	Easy
Electronic Devices	Middle	Normal	Possible
Desks Shelves	Low	Low	Difficult
Heated Grid Pattern	High	High	Easy

target feature points and to produce precise 3D reconstruction results.

The data set images include four different object types for reconstruction, as shown in Table II. The first object was a static (non-moving) human, which was the primary target to reconstruct primarily for future tracking processes. The human body temperature was relatively easy to detect and gave a higher resolution compared to other surrounding objects. The second object type were electronic devices that had relatively high temperatures depending on their power such as computers and the display monitors. The third type was the objects having low temperatures such as desk, shelves, and chairs that had a very low temperature and thermal signature in the images. These objects were difficult to detect and reconstruct by the IR O-D sensor. Another object was a heated grid pattern. It appeared as a sharp pattern due to its high temperature, so it gave us good results in order to find the position of the robot and the reconstructed points.

We chose imaging conditions of four types, a) long time, b) long path, c) large area and d) rotated path around the static target with respect to an indoor scale. As shown in Fig. 13, the first data set was taken over a long time, 60 sec, in a relatively small area and over a short distance. The second data set had a longer path of 3.2 m, in a small area. A longer data set was considered as the robot path increased with respect to the short path dataset for the indoor scale. We recorded the third image sequence in a larger area, about 80 m^2 , with a short path. This data set was used to represent a small scale outdoor environment by imitating vehicles and heat reflections

TABLE III
FOUR IMAGING CONDITIONS

Imaging Condition	Recording time (sec)	Robot path (m)	Covered area (m ²)
Long time	35	0.96	4.8
Long path	35	3.20	6.4
Large area	10	1.06	80
Rotated path	20	2.20	80
Average	26.66	1.74	30.4

using indoor objects. A larger outdoor scene had the maximum visibility of the area boundaries with thermal vision. The last data set includes a large area with a short path and the rotation would be toward the target when it is getting far from the robot. Table III shows the characteristics of the data sets for each image condition.

B. Reconstruction of Target Objects

The algorithm used a generic triangulation process for a reconstruction made from two images, which were taken in a large area data set. In our method, we used multiple images to find the camera world coordinates for each image sequence while using the triangulation process at the end of these sequences. The sum of the reconstruction error for each feature was calculated for reconstruction from two, three, five, ten and twenty observations. The reconstruction error in metric values was defined as the average of the difference between real distance and the calculated distance of the feature point. Our method updated the reconstructed points with respect to error and removed irrelevant points.

Fig. 14(c) shows the results for a generic triangulation method and a multi view reconstruction is given for four different objects. The difference among those objects was the number of feature points that were being tracked by the feature tracking process, given in Fig. 14(a). Reconstructed coordinates of different objects are depicted as different colors with respect to their distance in Fig. 14(b). The objects that are far away from the camera had an inconsistent and smaller amount of feature points than the others. This caused a tracking problem in the image sequence and a removal of the problem for the reconstructed points is giving a high error. Reconstruction error became smaller around 15.0-20.0% for the objects that had more feature points. The closest objects and middle distance objects are shown by dark and light blue in Fig. 14(c). Table IV also shows the objects that offer more and consistent feature points have around 15.0% error. The error percentages were calculated by using two distance values. We measured the real distance value, ground truth, in our experimental setup, and the calculated distance was obtained from the proposed 3D reconstruction method. It is shown in Fig. 14(a) that the heated grid pattern has more than 20 feature points and has around 15.0% reconstruction error

C. 3D Reconstruction Over Distance

With three different distances, in a large area data set, of objects were compared to see the efficiency of the reconstruction method pertaining to camera-object distance. Three different object distances with Blue represents close objects,

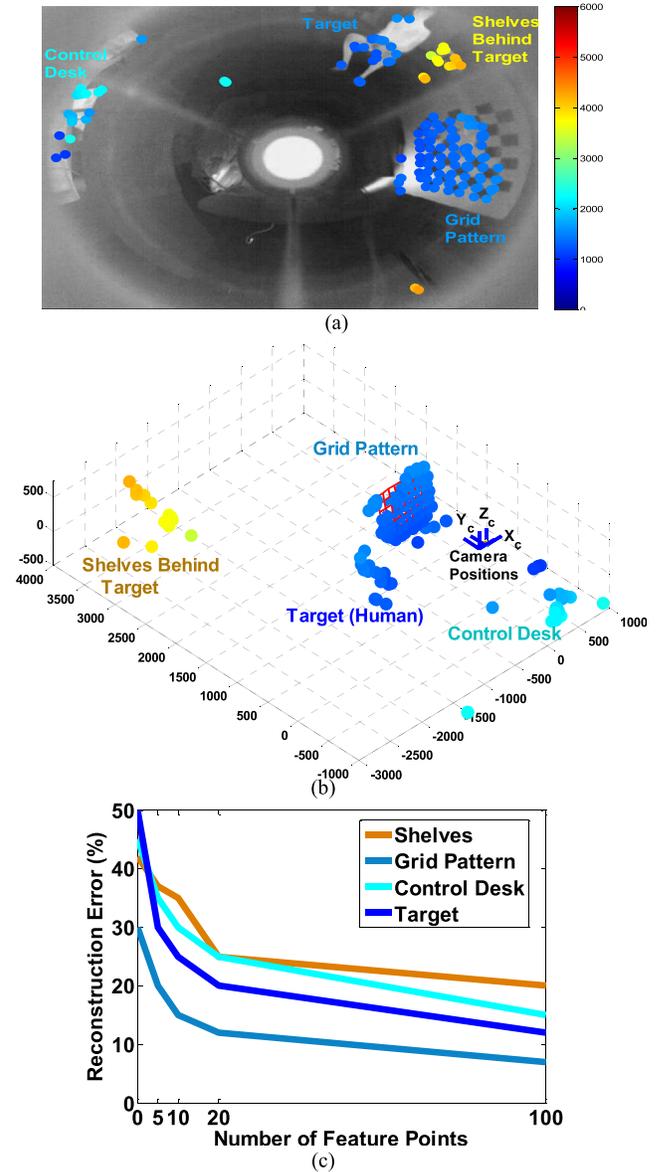


Fig. 14. (a) A depth map of the reconstructed scene from two images. The target feature points are shown in four colors; Shelves (Yellow), Grid pattern (Blue), Human (dark blue), and Control Desk (Light blue). (b) 3D plots in the world coordinates of those target points. (c) Reconstruction error for four representative target objects with respect to the number of feature points.

TABLE IV
TARGET RECONSTRUCTION FOR DIFFERENT OBJECTS

	Small	Middle	Large
Number of Feature Points	[1-10]	[11-20]	[21-100]
Average Reconstruction Error (%)	30.0	20.0	15.0

Green is for middle distances and Red is for objects far away from the camera. This is shown in Fig. 15(a).

We compared the reconstruction of the objects from three different distances and the average reconstruction error is given in Table V. If the objects were very close to or too far from the camera, reconstruction error would increase. We obtained the best result from the middle distance objects approximately 1 meter apart from the camera. The average reconstruction

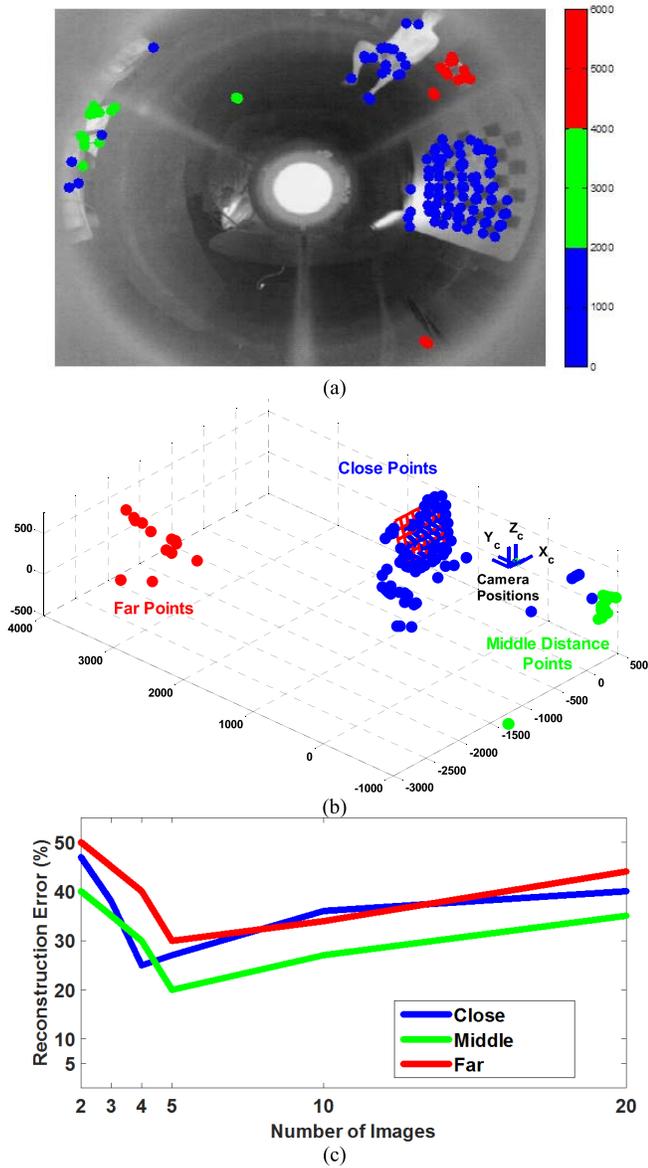


Fig. 15. (a) Depth mapping of the reconstructed scene from two images. The target feature points are shown in three colors; (Blue-close, Green-middle, and Red-far). (b) Reconstruction mapping the camera path based on the distance values. (c) Reconstruction error with respect to different distances (Blue-close, Green-middle, and Red-far) of the object position to the camera.

TABLE V
TARGET RECONSTRUCTION WITH DISTANCE

Metric	Close Distance	Middle Distance	Far Distance
Average Distance between the Target and Sensor (m)	0.5	1.0	3.0
Average Reconstruction Error (%)	30.0	20.0	40.0

error was increased to 40.0% for the objects set to a range of more than 3 meters to the camera position. These are shown by the red color in Fig 15(b). The average reconstruction error gave the lowest value of 20.0% for the middle distance objects after using the fifth and sixth images which corresponded to the fourth or fifth reconstruction process for the triangulation at those observation positions, Fig. 15(c).

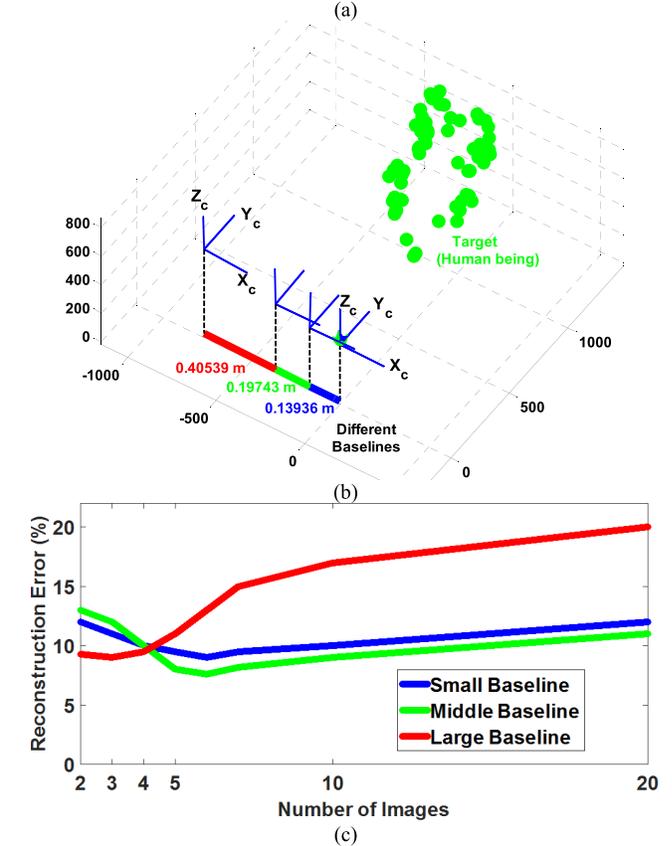
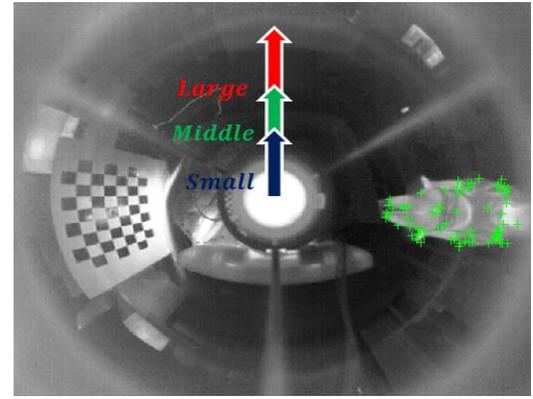


Fig. 16. (a) The feature points of the target on the original image. (b) Reconstruction map for the camera path with respect to different baseline distance between the camera positions, Red line shows the large baseline value Green line is the middle baseline value and Blue is the small baseline value. The feature points of the target are also shown with green. (c) Reconstruction error based on the different baseline values.

D. 3D Reconstruction Over Baselines

We compared three different baseline distances to find the minimum error for the reconstructed points in a large area data set. First, we selected a short distance of 0.139 m, to find the reconstructed points. Then, the baseline was increased to 0.197 m and finally 0.4 m to see the differences for those three different baselines. Our method used the values averaged from those distances for the next process to obtain the minimum reconstruction error.

A small baseline, shown in Fig. 16(b) by the blue trajectory, between 0.03 m and 0.15 m is visualized by a blue color path in Fig. 16(a). The middle baselines between

TABLE VI
TARGET RECONSTRUCTION WITH BASELINE

Metric	Small	Middle	Large
Average Distance of Baseline (m)	[0.03-0.15]	[0.15-0.25]	[0.25-1.00]
Average Reconstruction Error (%)	9.50	7.68	15.0

TABLE VII
NORMALIZED COST FUNCTION FOR NUMBER OF OBSERVATION POINTS

Data Set	2	3	4	5	6	7	10	15
Short path	1	0.56	0.50	0.44	0.38	0.38	0.49	0.47
Middle baseline	0.88	1	0.84	0.81	0.81	0.86	0.87	0.80
Size (MB)	0.31	0.46	0.62	0.78	0.93	1.09	1.56	2.34
Comp. time (s)	1.29	2.42	3.65	4.96	6.25	7.67	11.1	15.8

0.15 m and 0.25 m as well as large baseline distances greater than 0.25 m are visualized with green and red color paths in Fig. 16(b) and (a), respectively. For the small baseline case the average reconstruction error was 2.0% higher than for the middle baseline. The feature point matching the reconstruction became inefficient and missed the tracking of a few points when the baseline was 0.4 m or larger values. Thus, this high error increased the average error over 15.0% for the large baseline case since the target was lost in the view of camera. The best baseline range is at the middle range values, between 0.15 m to 0.25 m, which are shown in Table VI and Fig. 16(c). The best reconstruction error was for the middle baseline case with 5 to 7 images by 7.68% average value.

E. Optimal Image Number for Memory and Time Efficiency

The method in Section IV.E determined the optimal image number for each sequence. The optimal number of observations, n , was determined according to maximum accuracy, minimum error of reconstructed coordinates, maximum computational time, and system memory. This method obtained the minimum error when the system was working with full memory capacity as well as the maximum number of images possible to take along the trajectory. Each image had an average size of 0.156 MB, so the maximum number of images is given by $n = Size_{sys} / 0.156$. However, the time was the more important criterion for the reconstruction process.

The computational time for different quantities of images is given by Table VII. We considered the time to compute the 3D coordinates of each feature point from each image. The computational time was increasing almost linearly around 1 second for each image pair. The bold numbers give the minimum error for the common reconstruction error values of the datasets. The cost function values were derived from the average reconstruction error curve, computation time, and data size; given in the first two lines of Table VII with a minimum value of 6. The triangulation process gave the best value around the fifth triangulation using six observations, and we selected this value to be the optimal number for the reconstruction subset. Since, we obtained the best reconstruction error at that point, 6 images were used to get a minimum computational time and data size. The computational time for the 5th reconstruction step with 6 images was 6.25 seconds

TABLE VIII
DATA SET COMPARISON USING THE PROPOSED METHOD

Imaging Conditions	Long time	Long path	Large area	Rotated path	Average
Reconstruction error (%)	3.34	10.30	7.58	5.20	6.61
Reconstruction time (sec / data set)	18.7813	23.2004	7.7334	15.4669	16.2955

TABLE IX
OTHER SENSORY IMAGING USING THE PROPOSED METHOD (SECTION IV)

Average Result	Proposed IR O-D	Multiple Sensor (Color O-D) [9]	Stereo O-D [2]
Data Size (MB)	0.156 low	0.936 High	0.468 High
Reconstruction Time (s)	0.7	0.0014	0.35
Reconstruction Error (%)	10.64	8.44	7.51

and the data size was 0.93 MB with a minimum calculated error of 3.34%.

F. Comparison Among Data Sets and Other Methods

We compared the other data sets listed in Fig. 13 with the average reconstruction error and computational time. Table III also gives these four data sets. Table VIII gives the average reconstruction error and the average reconstruction time comparison for each data set. The reconstruction process was 18.7813 seconds for the long time data set and 15.4669 seconds for the rotated path data set. The reconstruction accuracy was sufficient with 3.34% and 5.20% respectively. The reconstruction time was faster with 7.7334 seconds for the large area data set; however, reconstruction error was 7.58% which is lower than the previous two methods. Reconstruction time was 23.2004 seconds for the long path with extended tracking; however, the reconstruction accuracy was the lowest, at 10.30% in Table VIII. We averaged the reconstruction error of four imaging conditions to get a 6.61%, and a reconstruction time of 16.2955 sec/data set and are shown in the Table VIII.

We compared the proposed method taking into consideration our sensory setting of the O-D IR camera, which has only one lower resolution sensor with respect to the compared works [2], [9], [22]. We compared the proposed method to the Color O-D + perspective camera setup, and the Stereo O-D camera setup in terms of the average reconstruction error, the required data size, and the computational time in the case of using only two observations of the proposed method. The three sensor settings were able to track fairly close in order to find the reconstructed pixel coordinates. The size of the data was small for the IR O-D case since the color information was not necessary and the image sizes were lower. The computational time was relatively lower for the color case and stereo case as they only require one observation from multiple sensors.

The IR O-D sensor had similar results as the other color camera and as the stereo methods, with a reconstruction error shown in Table IX. The proposed method gave very close results for average reconstruction error at 10.64%, to Color O-D Methods [9], and had a 3.13% higher error compared to the Stereo Method [2]. However, we increased the reconstruction accuracy by raising the number of observations.

TABLE X
METHODOLOGICAL COMPARISON TO OTHER ALTERNATIVES

Average Result	Method Proposed	Single Mirror [22]	Perspective + Color O-D [9]	Multiple Mirror [2]
Data Size	0.936 MB (6 IR images)	10.92 MB 70 images	0.936 MB 2 images	0.468 MB 1 image
Reconstruction Time (s)	6.25	18.0	0.0014	0.35
Reconstruction Error (%)	6.61	10.00	8.44	7.51

The minimally required data size based on Section VI.D, was 0.156 MB for IR O-D, which was 80.0% smaller than Color O-D + perspective and was 60.0% smaller than the Stereo O-D for the triangulation case. For the Stereo case the computational time process was completed in half of our proposed method's time since it uses only two images.

Four methods were compared in terms of the minimum data requirement for the minimum error result. The proposed method was 0.90% better than the average reconstruction error and 7.51% better than the Multiple Mirror Method [2]. It was 1.83% better than the Color O-D Method [9] and was 3.39% better than the Single Mirror Method [22]. The data size for the proposed method was around 90.0% smaller than [22] and was 50.0% larger than the average data size of [2]. The average reconstruction time of the Single Sensor Method was 18 seconds, which is longer than the proposed method by 7.67 seconds; however, the Multiple Sensor method had a shorter reconstruction time of 0.35 seconds. Concerning the data size required for these reconstruction methods, the proposed method used an average of six IR images and 0.936 MB of data size for reconstruction, which was significantly smaller than the single mirror method and almost equivalent with other methods, and reconstruction error improved, shown in Table X. The sensory setting of the proposed method increased, which in turn extends the conditions for use (daytime and night time) in addition to improved 3D reconstruction results in spite of having lower resolution and less camera sensors than compared works.

VI. CONCLUSION

We proposed Dynamic 3D Reconstruction (D3DR) that used correlation based detection and corner feature detection methods to find the best feature points in noisy thermal O-D images. We first used the target detection method to find the central position of the objects and determined the target area. Then we used the second method to detect more intense feature points for the reconstruction of the low noise region. The best number of observations, the baseline distance between observations, and the distance from the target region were also calculated in the experimental section to improve the results of 3D reconstruction. We were able to show that the comparison of the results between the proposed IR O-D and the color O-D camera was on average 1.38% improvement in the reconstruction error while being 3.39% better than the Single Mirror Method; even though 90.0% less data size was required compared to other studies and it took approximately an equal amount of computational time. We plan to track target

trajectories using the 3D reconstruction results and recalculate the distance of the target region as a future work of this research.

ACKNOWLEDGMENT

The authors would like to thank Fullah Fanary and Jeff Miller for proofreading to help improve this paper.

REFERENCES

- [1] G. Caron, E. M. Mouaddib, and E. Marchand, "3D model based tracking for omnidirectional vision: A new spherical approach," *Robot. Auton. Syst.*, vol. 60, no. 8, pp. 1056–1068, 2012.
- [2] Z.-H. Xiong, I. Cheng, W. Chen, A. Basu, and M.-J. Zhang, "Depth space partitioning for omni-stereo object tracking," *IET Comput. Vis.*, vol. 6, no. 2, pp. 153–163, Mar. 2012.
- [3] D. Um, "Multiple intensity differentiation-based 3-D surface reconstruction with photometric stereo compensation," *IEEE Sensors J.*, vol. 14, no. 5, pp. 1453–1458, May 2014.
- [4] M. G. Diaz, F. Tombari, P. Rodriguez-Gonzalvez, and D. Gonzalez-Aguilera, "Analysis and evaluation between the first and the second generation of RGB-D sensors," *IEEE Sensors J.*, vol. 15, no. 11, pp. 6507–6516, Nov. 2015.
- [5] U. Rehman and S. Cao, "Augmented-reality-based indoor navigation: A comparative analysis of handheld devices versus Google glass," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 1, pp. 140–151, Feb. 2017.
- [6] H. Lu, S. Yang, H. Zhang, and Z. Zheng, "A robust omnidirectional vision sensor for soccer robots," *Mechatronics*, vol. 21, no. 2, pp. 373–389, 2011.
- [7] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera," *IEEE Trans. Robot. Autom.*, vol. 16, no. 6, pp. 890–898, Dec. 2000.
- [8] D. Scaramuzza and R. Siegwart, "Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1015–1026, Oct. 2008.
- [9] M. Lauer, M. Schönbein, S. Lange, and S. Welker, "3D-objecttracking with a mixed omnidirectional stereo camera system," *Mechatronics*, vol. 21, no. 2, pp. 390–398, 2011.
- [10] P. Robinette, A. M. Howard, and A. R. Wagner, "Effect of robot performance on human-robot trust in time-critical situations," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 4, pp. 425–436, Aug. 2017.
- [11] D. Um, D. Ryu, and M. Kal, "Multiple intensity differentiation for 3-D surface reconstruction with mono-vision infrared proximity array sensor," *IEEE Sensors J.*, vol. 11, no. 12, pp. 3352–3358, Dec. 2011.
- [12] S. Vidas, P. Moghadam, and M. Bosse, "3D thermal mapping of building interiors using an RGB-D and thermal camera," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2013, pp. 2311–2318.
- [13] C. Liu and Y. Zhang, "3D trajectory reconstruction from monocular vision based on prior spatial knowledge," *IEEE Sensors J.*, vol. 16, no. 3, pp. 817–822, Feb. 2016.
- [14] J. Feng, Y. Wang, and S. F. Chang, "3D shape retrieval using a single depth image from low-cost sensors," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.
- [15] H. Aliakbarpour, J. F. Ferreira, K. Palaniappan, G. Seetharaman, J. Dias, and V. B. S. Prasath, "A probabilistic fusion framework for 3-D reconstruction using heterogeneous sensors," *IEEE Sensors J.*, vol. 17, no. 9, pp. 2640–2641, May 2017.
- [16] L. Puig, J. Bermúdez, P. Sturm, and J. J. Guerrero, "Calibration of omnidirectional cameras in practice: A comparison of methods," *Comput. Vis. Image Underst.*, vol. 116, no. 1, pp. 120–137, 2012.
- [17] A. Torii, M. Havlena, and T. Pajdla, "Omnidirectional image stabilization for visual object recognition," *Int. J. Comput. Vis.*, vol. 91, no. 2, pp. 157–174, 2010.
- [18] R. Kawanishi, A. Yamashita, and T. Kaneko, "Estimation of camera motion with feature flow model for 3D environment modeling by using omni-directional camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2009, pp. 3089–3094.
- [19] R. Kawanishi, A. Yamashita, and T. Kaneko, "Three-dimensional environment model construction from an omnidirectional image sequence," *J. Robot. Mechatron.*, vol. 21, no. 5, pp. 574–582, 2009.
- [20] D. Ishizuka, A. Yamashita, R. Kawanishi, T. Kaneko, and H. Asama, "Self-localization of mobile robot equipped with omnidirectional camera using image matching and 3D-2D edge matching," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Nov. 2011, pp. 272–279.

- [21] R. Bunschoten and B. Krose, "Robust scene reconstruction from an omnidirectional vision system," *IEEE Trans. Robot. Autom.*, vol. 19, no. 2, pp. 351–357, Apr. 2003.
- [22] C. Mei, S. Benhimane, E. Malis, and P. Rives, "Efficient homography-based tracking and 3-D reconstruction for single-viewpoint sensors," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1352–1364, Dec. 2008.
- [23] Y. Bastanlar, A. Temizel, Y. Yardimci, and P. Sturm, "Multi-view structure-from-motion for hybrid camera scenarios," *Image Vis. Comput.*, vol. 30, no. 8, pp. 557–572, 2012.
- [24] A.-E. Nagy, I. Szakats, T. Marita, and S. Nedeveschi, "Development of an omnidirectional stereo vision system," in *Proc. IEEE Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, Sep. 2013, pp. 235–242.
- [25] M. Gutin, E. K. Tsui, O. Gutin, X.-M. Wang, and A. Gutin, "Thermal infrared panoramic imaging sensor," *Proc. SPIE*, vol. 6206, p. 62062E, May 2006.
- [26] L. Zhang, Y. Hu, J. Zhang, and Y. Li, "Multiview geometry in traditional vision and omnidirectional vision under the ∞ -norm," *IET Comput. Vis.*, vol. 6, no. 1, pp. 13–20, Jan. 2012.
- [27] C. Mei and P. Rives, "Single view point omnidirectional camera calibration from planar grids," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 2007, pp. 3945–3950.
- [28] F. Fraundorfer, D. Scaramuzza, and M. Pollefeys, "A constricted bundle adjustment parameterization for relative scale estimation in visual odometry," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2010, pp. 1899–1904.
- [29] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *Proc. IEEE Int. Conf. Comput. Vis. Syst. (ICVS)*, Jan. 2006, p. 45.
- [30] T. Svoboda and T. Pajdla, "Epipolar geometry for central catadioptric cameras," *Int. J. Comput. Vis.*, vol. 49, no. 1, pp. 23–37, 2002.
- [31] A. Akin, E. Erdede, H. Afshari, A. Schmid, and Y. Leblebici, "Enhanced omnidirectional image reconstruction algorithm and its real-time hardware," in *Proc. 15th Euromicro Conf. Digit. Syst. Design (DSD)*, Sep. 2012, pp. 907–914.
- [32] W. K. Wong, C. K. Loo, and W. S. Lim, "Optical Approach Based Omnidirectional Thermal Visualization," *Int. J. Image Process.*, vol. 4, no. 3, pp. 263–286, 2010. [Online]. Available: <http://www.cscjournals.org/library/manuscriptinfo.php?mc=IJIP-120>
- [33] Wahyono, H. Joko, A. Vavilin, and K.-H. Jo, "A geometry-based 3D reconstruction from a single omnidirectional image," in *Proc. 19th Korea-Jpn. Joint Workshop Frontiers Comput. Vis. (FCV)*, Jan./Feb. 2013, pp. 295–299.
- [34] R. Kawanishi, A. Yamashita, T. Kaneko, and H. Asama, "Parallel line-based structure from motion by using omnidirectional camera in textureless scene," *Adv. Robot.*, vol. 27, no. 1, pp. 19–32, 2013.
- [35] L. Bagnato, P. Frossard, and P. Vanderghenst, "A variational framework for structure from motion in omnidirectional image sequences," *J. Math. Imag. Vis.*, vol. 41, no. 3, pp. 182–193, 2011.
- [36] W. Chen, M. Zhang, and Z. Xiong, "Nonsingle viewpoint omni-stereo depth estimation via space layer labeling," *Opt. Eng.*, vol. 50, no. 4, p. 047005, 2011.
- [37] T. Mouats, N. Aouf, L. Chermak, and M. A. Richardson, "Thermal stereo odometry for UAVs," *IEEE Sensors J.*, vol. 15, no. 11, pp. 6335–6347, Nov. 2015.
- [38] A. A. Altahir *et al.*, "Optimizing visual sensor coverage overlaps for multiview surveillance systems," *IEEE Sensors J.*, vol. 18, no. 11, pp. 4544–4552, Jun. 2018.
- [39] R. I. Hartley, "In defence of the 8-point algorithm," in *Proc. 5th Int. Conf. Comput. Vis.*, vol. 1995, pp. 1064–1070.
- [40] *Computer Vision System Toolbox Documentation*. Accessed: Feb. 7, 2017. [Online]. Available: <https://www.mathworks.com/help/vision/>
- [41] RemoteReality. *Intelligent Awareness*. Accessed: Jul. 8, 2016. [Online]. Available: <http://www.remotereality.com/>
- [42] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2006, pp. 5695–5701.



Emrah Benli (S'15–M'17) received the B.Sc. degree in electronics and telecommunication engineering from Kocaeli University, Kocaeli, Turkey, in 2009, the M.Sc. degree in electrical and computer engineering from Clemson University, Clemson, SC, USA, in 2013, and the Ph.D. degree in electrical and computer engineering from Virginia Commonwealth University, Richmond, VA, USA, in 2017. He is currently a Post-Doctoral Fellow specializing in autonomous mobile robotics with the Computational and Information Sciences Directorate, U.S. Army Research Laboratory, Adelphi, MD, USA.

His research interests include intelligent systems, computer vision, artificial intelligence, multi-modal sensory, robotic system design and control, and human–robot interaction.



Jonas Rahlf received the B.Sc. and M.Sc. degrees in systems engineering from the University of Bremen, Germany, in 2016. He is currently an Embedded Systems Engineer in the field of contactless distance measurement with trinamiX GmbH, Ludwigshafen, Germany.

His current research interests include laser rangefinders (LIDAR), intelligent systems, and autonomous vehicles.



Yuichi Motai (S'00–M'03–SM'12) received the B.Eng. degree in instrumentation engineering from Keio University, Tokyo, Japan, in 1991, the M.Eng. degree in applied systems science from Kyoto University, Kyoto, Japan, in 1993, and the Ph.D. degree in electrical and computer engineering from Purdue University, West Lafayette, IN, USA, in 2002. He is currently an Associate Professor of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA, USA.

His research interests include the broad area of sensory intelligence, particularly in medical imaging, pattern recognition, computer vision, and sensory-based robotics.