Two-layer Tracking for Occlusion Handling and Inter-sensor Identification in Multiple Depth Sensors-based Object Detection and Tracking

Houari Sabirin
Ultra-realistic Communications Group
KDDI Research, Inc.
Fujimino-shi, Saitama, Japan
ho-sabirin@kddi-research.jp

Sei Naito
Ultra-realistic Communications Group
KDDI Research, Inc.
Fujimino-shi, Saitama, Japan
sei@kddi-research.jp

***Abstract*—The main challenge in depth-based object detection and tracking process is to provide correct identification of the detected objects during occlusion. This is because the information necessary to distinguish and consequently identify the objects throughout the occlusion events are limited, compared to conventional, color-based object tracking. In this paper we propose a two-layer tracking method that enables automatic occlusion handling and inter-sensor identification for object detection and tracking that utilizes more than one depth sensor. On the first layer, the tracking is first performed independently for each sensor to extract objects' feature and perform initial tracking with separation of the occluded objects. On the second layer, the tracking is performed in the perspective projection of the objects tracked on the first layer that are combined in a single processing plane to provide correct identification of the objects that are detected in one sensor to another. Experiment results show that the proposed method can correctly identified occluded objects and objects that are moving between sensors coverage area.**

Keywords—surveillance; object detection and tracking; depth data; depth-based separation; range sensor

I. INTRODUCTION

In some countries the installation of surveillance camera should be made known to the people inside the monitoring area so that they are aware that their behaviors are being recorded. However, there are situation where people tend to feel uncomfortable being watched by the camera. Some issues regarding possible privacy infringements from surveillance camera were discussed in [1]. One solution for this issue would be to install depth sensor camera that can detect the moving objects and measure the distance of the objects from the sensor. By only "seeing" the distance of the objects, the sensor cannot recognize the actual and detailed texture of the detected objects (such as people face or their clothing textures) thus would be sufficient to protect the privacy. Furthermore, by sensing the distance, moving objects can still be recognized regardless of lightning conditions of the monitored area.

Due to only sensing the distance, one of challenging topics in moving object detection and tracking research is to enable accurate tracking of the objects during occlusion [2]. While conventional (i.e., RGB camera) object detection and tracking system has virtually abundant information that can be extracted from color signals, in depth-based object tracking we practically only have the depth signal. Technically, a depth sensor measures the distance of infra-red light signals between the sensor and the captured object. In some cases, light interferences affect measurement stability of the distance taken at certain point in the captured frame. Although some sensors have a de-noising feature to reduce such noise by setting the signal amplitude threshold, yet setting the threshold to reduce noise may cause a reduction in coverage range.

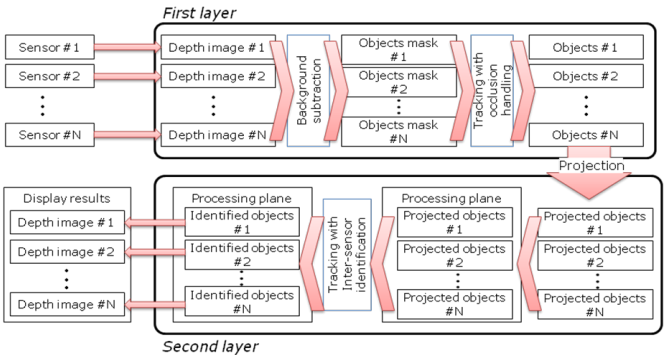

Fig. 1. Overall flow of the two-layer tracking method

In order to achieve satisfactory object tracking result with such restrictions, we propose a method to correctly and uniquely identify objects even during occlusion. Moreover, to enable the utilization of depth sensor in larger coverage area, we develop an inter-sensor identification method so that when more than one depth sensors are utilized, the detected objects that are moving from one sensor to another can still be correctly identified.

Our contribution can be explained in brief by the flow of the proposed two-layer tracking method as illustrated in Fig. 1. On the first layer, the depth images are acquired from the sensors, where background subtraction is first performed to obtain the objects masks and define objects features so that initial tracking in each sensor with occlusion handling can be performed using those features. Next, the second layer receives perspective projections of positions of objects from the first layer and combine these projected positions onto

ISBN 978-0-9928626-7-1 © EURASIP 2017

single processing plane. The tracking with inter-sensor identification is then performed to correctly identified the objects that move between the sensors so that no duplicated objects exist. Lastly, the final tracking results are shown by re-projecting the correctly identified objects position onto the depth image.

We will briefly revisit related depth-based object detection and tracking works in Section 2. Section 3 provides more detailed discussions of our proposed method followed by its experimental results and analysis in Section 4. This paper is concluded in Section 5.

II. Related Work

The benefits of utilizing depth in human motion analysis have been shown for many years in many applications such as depth-based activity recognition, head and face detection, etc. [4]. Identifying detected objects in RGB-D camera-based object detection and tracking using both color and depth image have been proven to produce accurate estimation of occluded moving objects during occlusion. A combination of the Lucas-Kanade method for compatibility with conventional image processing and the inverse compositional method for better computational performance is proposed in [5]. In [6], the depth scaling kernelised correlation filters (DS-KCF) tracker utilizes KCF in an RGB image to update the segmented target region in the depth data as a scale guide to update the target's model.

In some studies, 3D images synthesized from stereo cameras, which are not uncommon for object tracking [7], are also utilized to support feature extraction to generate image sets for training data in object tracking [8], as well as utilizing the Bhattacharyya distance to find the similarity between two probability density functions of a cluttered depth map from the 3D image [9]. ViBe, a method that estimates the background via a sample-based random selection policy of the background values for color image is proposed in [10]. It applies the method for depth data acquired from a stereo camera. Particle filtering, which is one of the common methods in object tracking, is employed in [11] with active contours to estimate the global motion of the objects in tracking moving vehicles based on 3D range data.

Some research also utilizes range sensor cameras without RGB signals, where the nonexistence of the color image makes it impossible to use color as feature for object tracking. Depth-only image analysis using such cameras has been presented in [12] to perform single object tracking to track selected single static objects (such as a face, cup, toys, etc.) and correctly identify the object even when the object is occluded with moving subjects (such as moving hands or books). Object detection and tracking system without occlusion handling using depth-only image was developed in [13], while detection and tracking of multiple objects with occlusion detection but no identification during the occlusion were presented in [14] and [15]. Our proposed depth-based object detection and tracking method enables individual object identification for each object during occlusion. This feature has not been provided in any depth-based object detection and tracking methods we reviewed.

III. Two-layer Tracking

A. First Layer

The process on the first layer is performed for each sensor. It is started by acquiring the depth image from the sensor and obtains the depth values from the image and is independent between sensors. The left part of Fig. 2 illustrates the process performed on the first layer for the case where two sensors that are facing toward each other are utilized. This means that the area roughly from the half part to the upper part of the depth image from *Sensor*#1 would correspond to the same area of depth image from *Sensor*#2. From the depth image, we acquire its depth value for each pixel that represents the distance (in 10^2 millimeter) of the corresponding points in the monitored area. The pixels in the area beyond the coverage of the sensor are marked with zero values, thus shown as black pixels in the depth value's gray-level image.

The background subtraction method is performed based on [3] where statistical information of the depth data is utilized to remove the background by first creating a background model. To do this, a *Background* sequence was created to capture the monitored area without any moving objects. A threshold for each pixel in this sequence is defined by calculating the mean and variance of depth value throughout the sequence.

Let $D(i,j)_f$ be a depth value at position i, j in the f-th frame, the depth data is then classified into background (zero) and non-background (one) by the following conditions

$$D(i,j)_f = \begin{cases} 0, & \min\left(\hat{\mathbf{d}}\right) - \alpha \max\left(\Delta\mathbf{d}\right) < D(i,j)_f \\ & < \max\left(\hat{\mathbf{d}}\right) - \beta \max\left(\Delta\mathbf{d}\right) \\ 1, & otherwise \end{cases} \quad (1)$$

where $\alpha = 1$ and $\beta = 3$ are set based on experiments. Here $\hat{\mathbf{d}}$ is defined as the set of depth value that satisfies $\left|D(i,j)_f - \lambda(i,j)\right| \le 2\sigma(i,j)$ in all F frames, and $\Delta\mathbf{d}$ as the set of depth value differences for each pixels between two consecutive frames given by $\Delta d(i,j) = \left|D(i,j)_f - D(i,j)_{f-1}\right|$. The mean λ and variance σ are calculated from the *Background* sequence as $\lambda(i,j) = \frac{1}{F}\sum_{f=0}^{F} D(i,j)$ and $\sigma(i,j) = \frac{1}{F}\sum_{f=0}^{F}\left(\lambda(i,j) - D(i,j)_f\right)^2$, respectively.

From the background-subtracted depth image, *object masks* are defined as groups of pixels where each of them represent a moving object (or a candidate of moving object, as one mask may also comprises more than one objects when occlusion occurs). Let O_m^f be the m-th object mask at the f-th frame, we define its features: position, size, depth values, and its identification label as $\mathbf{P}_m^f = \{i,j\}$, $\mathbf{S}_m^f = \{w,h\}$, $\mathbf{D}_m^f = \{d_{w-i,h-j},\ldots,d_{w+i,h+j}\}$ and ℓ_m^f, respectively. Position i, j is determined as the center of the object mask of size w width and h height. Thus the depth values are assigned from the top-left rectangular point of the object mask, to its bottom-right point. Note that since the process on the first layer is performed independently for each sensor, we reduce the

ISBN 978-0-9928626-7-1 © EURASIP 2017

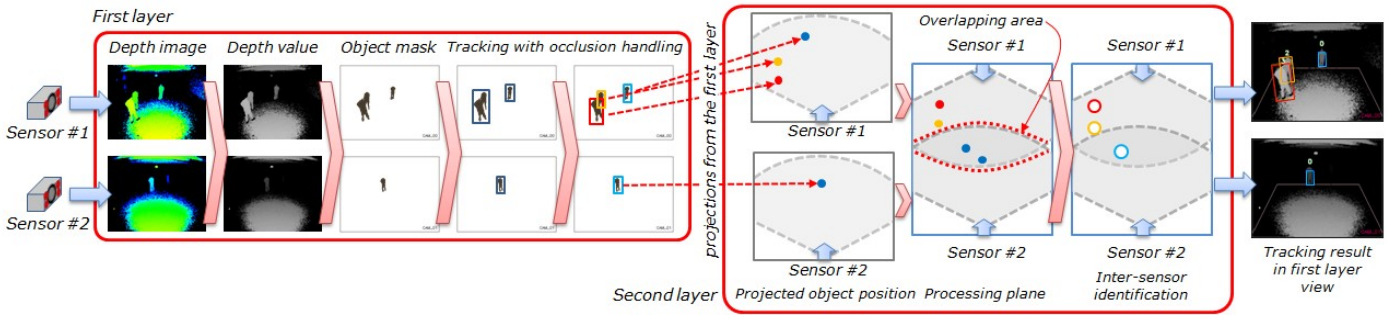

Fig. 2. Illustration of detection and tracking on the first layer and second layer involving two sensors.

complexity of managing the objects' features by avoiding the correlation of objects among the sensors.

Utilizing the position feature $\mathbf{P}_m^f$, initial object tracking on the first layer is performed by employing a similarity matching between the positions of two objects in two consecutive frames. More specifically, the *m*-th object in frame *f* is assigned the same identity as the *n*-th object in frame *f*-1 if the following rule is satisfied

$$n_{match} = \arg\min_n \left(\left\| \mathbf{P}_m^f - \mathbf{P}_n^{f-1} \right\| \right). \quad (2)$$

However, when occlusion occurs, similarity matching based on position cannot be simply utilized until the occluded objects have been separated. Therefore we perform automatic occlusion detection to recognize when two or more objects are occluded with each other. Next, separation is performed so that similarity matching can be applied onto each object as independent entities.

Automatic occlusion detection is performed as illustrated in Fig. 3(a). Firstly, the area of an object A_m^f is calculated by simply multiplying width and height of object's size feature $\mathbf{S}_m^f$. Next, to determine an occlusion, we observe the areas of more than one moving object in the previous frame that intersect with another object in the current frame. Suppose an intersected area produced when A_1^{f-1} (the area of O_1^{f-1}) and A_2^{f-1} (the area of O_2^{f-1}) intersect with A_1^f (the area of O_1^f) as illustrated in Fig. 3(a). To determine that the intersect indicates an occlusion, two conditions apply: 1) the intersect areas of A_1^{f-1} and A_2^{f-1} with A_1^f, denoted by $\tilde{A}_{1,1}^{f-1}$ and $\tilde{A}_{1,2}^{f-1}$ respectively, shall be larger than a given threshold; and 2) at least two areas are intersecting between the current object and any objects in the previous frame. To avoid the possibilities of depth masks that are not actual moving objects being perceived as occluded objects, an intersect area is determined to be larger than 5% of the area of A_n^{f-1}. According to the above conditions, the example in Fig. 3(b) does not considered as occlusion.

After an occlusion has been detected, the occluded objects are separated according to the depth values feature $\mathbf{D}_m^f$ to enable separated tracking of each occluded object. Let $\tilde{\mathbf{P}}_1^f = \{\tilde{\mathbf{p}}_{1,1}, \ldots, \tilde{\mathbf{p}}_{1,K}\}$ be the center points of intersection areas

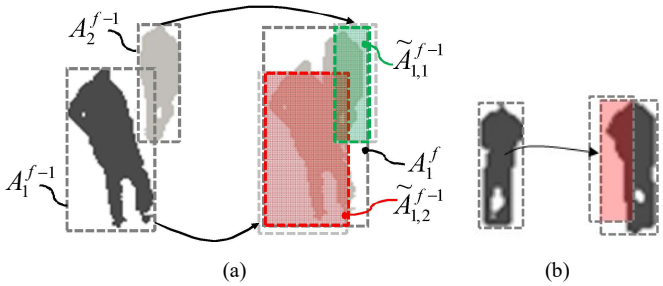

Fig. 3. (a) An example of occlusion detection between objects in current frame (right) and previous frame (left). (b) Example of intersecting area with only one object involved thus does not considered as occlusion.

$\mathbf{A}_1^f = \{\tilde{A}_{1,1}^{f-1}, \ldots, \tilde{A}_{1,K}^{f-1}\}$, we first assign initial label to each center point. A classifier is then performed to assign the same label for all pixels with similar depth value as the depth value in each center points. The similarity measurement

$$\Delta\delta_{u,v} = \left| \tilde{d}_{i,j} - \tilde{d}_{u,v} \right| \forall i-1 \le u \le i+1, j-1 \le v \le j+1 \quad (3)$$

is computed so that all pixels with $\Delta\delta_{u,v} < \delta_{Tresh.}$ would be assigned the same label as the center point. Here $\delta_{Tresh.}$ is set heuristically to 128 millimeters.

Finally, the features of object mask of the occluded object is updated according to the labeled pixels. That is, its position and size are now adjusted to the new separated positions and sizes of group of pixels with the same label. By now, the occluded objects are now separated and each of them can be tracked individually with similarity matching in (2).

B. Second Layer

On the second layer, the tracking is performed for the perspective projection of the positions of objects defined on the first layer. The perspective transform matrix is determined by selecting reference points from the floor area of the depth sensor view, which was performed in the *Background* sequence. Thus the second layer represents the foot position of the object mask.

To handle multiple sensors tracking on this layer, the projected points from all sensors are combined onto one *processing plane*. The perspective projection in the processing plane takes into account relative positions of one sensor to another. For example, as shown in the right part of Fig. 4 for the case when two sensors are facing towards each other, the projection of positions from one sensor would be followed by

ISBN 978-0-9928626-7-1 © EURASIP 2017

flipping the points so that the points in the processing plane correctly represent a single monitoring area (e.g. for two sensors facing toward each other, the upper part of processing plane represent the objects from the first sensor while the objects from the other sensor is at the lower part of the plane). Here, an *overlapping area* in the processing plane is defined to indicate the area between two sensors that overlap with each other, which is determined to be no larger than ¼ of each sensor's area. This area is utilized to determine whether the detected objects in one sensor are also detected in the other sensors and perform inter-sensor identity matching. This will ensure that the tracked object in one sensor that moves to the area covered by another sensor is correctly identified as the same object.

In contrast to the first layer, on the second layer we dealt only with the projected positions and the identity of the objects. That is, neither depth information or object's shape and size are taken into account in the tracking on this layer. Additionally a label to indicate the source of the objects is added. Thus we define ${}_c\hat{O}_m^f$ as the *m*-th object at the *f*-th frame on the second layer (i.e. in the processing plane) that is projected from the *c*-th sensor in the first layer, with features ${}_c\hat{\mathbf{P}}_m^f$ as its position and ${}_c\hat{\ell}_m^f$ as its assigned label.

At the first frame, object's labels on the second layer are initially determined by their relative positions from top-left point of the frame, regardless the identification given that has been assigned on the first layer. In the succeeding frames, the identification is determined by position feature matching with the same calculation as in (2). Since occluded objects have been handled by separation on the first layer, it is ensured that there are no more occlusion occurred on the second layer; thus position feature matching would correctly identify the objects throughout the sequence, except inside the overlapping area. In the overlapping area, objects from sensor are assumed to be also detected in another sensor. In this area, the position features of two or more objects from different source (i.e. detected from different sensors) are expected to be entangled with each other, creating duplicated projected position of the same object from different sensors. Therefore we remove the duplicated information of the objects' positions when objects from different sensors are detected inside the overlapping area.

Let $\boldsymbol{\Theta}$ be the overlapping area, similarity matching derived from (2) is then performed if ${}_c\hat{\mathbf{P}}_m^f \cap \boldsymbol{\Theta}$, to satisfy

$$n_{dupl} = \arg\min_n \left(\left\| {}_c\hat{\mathbf{P}}_m^f - {}_b\hat{\mathbf{P}}_n^f \right\| < \hat{\delta}_{Tresh} \right) \tag{4}$$

given $c \neq b$. Accordingly, object with position index conforms to n_{dupl} will then be denoted as duplicated entry and its identity will be assigned as ${}_b\hat{\ell}_n^f \leftarrow {}_c\hat{\ell}_m^f$. Here, only one object among the duplicated objects with the same identity is counted for further tracking process. In (4), similarity threshold is defined as the spatial likelihood between two duplicated entries detected from different sensors given by $\hat{\delta}_{Tresh} = ¼H - \left\| {}_c(P_\theta)_m^f - {}_b(P_\theta)_n^f \right\|$ where H is the length of longer side of the processing plane and ${}_c(P_\theta)_m^f$ is the horizontal ($\theta = i$) or vertical ($\theta = j$) component of object's position in the processing plane, depends on how the images of the first layer is arranged in the processing plane: horizontally (side by side) or vertically (top and above). Note that regardless the number of sensors utilized, the calculation of matching within overlapping area is conducted for a pair of sensors for simplification.

IV. Experimental Results

To our knowledge, there are no publicly available datasets produced by multiple depth sensors that provide information of depth value in each pixel of the frame. Therefore we create our own test sequences to implement the proposed method. As aforementioned, an overlapping area in the processing plane applies to the overlapping area between pair of sensors. Thus to implement and justify the tracking between multiple sensors it would be sufficient to conduct the experiments using two sensors, for the sake of simplicity, without compromising the purpose of the proposed method.

We set experimental setup as implied in Fig. 2 where two depth sensors are located at each end of a room that roughly 15 meters apart, located at 2.6 meters from the ground with tilt angle of around 45 degrees, and around 70 degrees of field of view. To avoid light interference between sensors, the maximum coverage distance of each of them is set differently to 8 meters and 7 meters, respectively (each sensor has maximum coverage of around 13 meters but it performs best within 7-8 meters). These settings will produce an overlapping area from both sensors at the center of the room. From this room setup, a 250 frames *Background* sequence was created for each sensor.

The depth images acquired from the sensors are in QVGA resolutions (320×240) with 25fps of frame rate. The two sensors are connected to a PC via network cable where the depth images and the depth data are transmitted simultaneously to produce two sets of depth image and depth data from the two sensors without significant time delay between them. In the experiments we first store the depth images and depth data into storage and perform offline processing afterwards. The computation of depth values is performed at the beginning of the processing of each frame. We took around 8000 frames with different objects movement to test our method.

Table I summarizes the performance of the occlusion detection and inter-sensor identifications (two-layer tracking) for several test sequences. The percentage values indicate the ratio of the number of frames where two-layer tracking correctly performed against the ground truth (the number of two-layer tracking that are manually annotated). Here, up to 99% accuracy can be achieved for occlusion tracking and 94% accuracy for inter-sensor tracking. The performance of the proposed method depends on the movements of objects and their positions with each other. From the experiments we found that the occlusion detection works better when the occluded objects are located within the center part of the coverage area; when an occlusion occurs around the overlapping area, the saliency of the depth values of the

ISBN 978-0-9928626-7-1 © EURASIP 2017

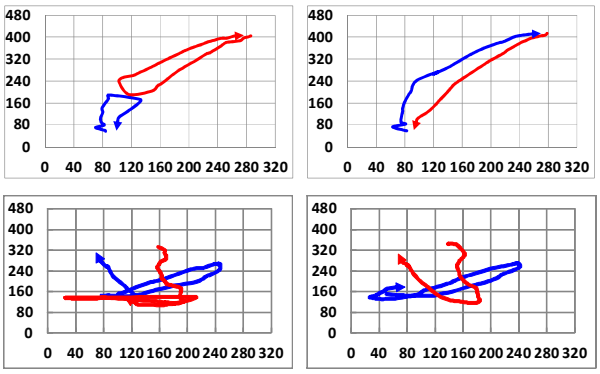

Fig. 5. Plots of trajectories of detected objects (left) without and (right) with occlusion tracking. Horizontal and vertical axes represent the width and the height of processing plane in pixels, respectively.

detected objects may reduce the accuracy of the occlusion detection. We also observe from the experiments that occlusion handling works better if the distance between the occluded objects, in terms of computed depth values, is larger than 200 millimeters. The effect of occlusion tracking can be seen in the plots of the objects' trajectories from the processing plane in Fig. 4. Without occlusion tracking as in [14] and [15], the identity of occluded objects cannot be retained after the occlusion. Inter-sensor identification depends on the pre-defined settings for determining the overlapping area. For $\pm$15 meters area from two sensors with each 8 meters and 7 meters optimal coverage range, intersecting one-third of area from projected floor plane of each sensor is sufficient to correctly identified most of inter-sensor movements of objects.

There are some failures occur when depth values are over-filtered especially in the overlapping area where the object mask cannot be defined on the first layer; thus the object information that was present in previous frame is missing in current frame. As shown in Table I, even with high accuracy of occlusion tracking in one of the sensor, the 46% accuracy in *Sequence* #3 is due to more occlusions occurred within the overlapping area. Especially, in this area *Sensor* #1 cannot maintain good tracking performance. Consecutively, the object information is also missing on the second layer. As a result, if the depth values of this object is re-appeared on the succeeding frame, incorrect identification may occurs. Currently, this problem can be solved by extending the computation of similarity matching in (2) into a predefined duration. In the experiment we select up to 6 frames of sliding observation window for the algorithm to find corresponding object at the same position before the depth information is missing. Nevertheless, when too many objects are close to each other during this period, the extended similarity matching may also failed.

V. Conclusions

We have introduced a method that enable identification of detected objects in the event of occlusions and inter-sensor identification for objects that are moving between sensors coverage area. These are achieved by utilizing depth information in two steps: initial tracking and occlusion handling in sensor world coordinate, followed by inter-sensor tracking in a combined processing plane by performing perspective projection of the detected objects' positions. While the results show satisfactory tracking accuracy, in more complicated occlusion scene (e.g. more than three objects walk closely with many of overlaps) our works will need further improvements.

TABLE I. Performance of Two-Layer Tracking

Seq.	Frames	Obj.	Occlusion tracking		Inter-sensor tracking
			Sensor #1	Sensor #2	
1	3110	3	61.63%	89.46%	85.03%
2	3000	3	99.20%	84.73%	94.49%
3	2230	4	39.73%	94.49%	46.99%

References

[1] American Civil Liberties Union, What's Wrong With Public Video Surveillance?, https://www.aclu.org/technology-and-liberty/whats-wrong-public-video-surveillance

[2] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," IEEE Transaction on Systems, Man, and Cybernetics – Part C: Application and Reviews, vol. 34, no. 3, pp. 334-352, August 2004.

[3] S.H. Cho, K. Bae, K.M. Kyung, S. Jeong, and T.C. Kim, "Background subtraction based object extraction for time-of-flight sensor," in Proc. IEEE 2nd Global Conf. on Consumer Electronics, pp. 48-89, 2013.

[4] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall, "A survey on human motion analysis from depth data," Time-of-Flight and Depth Imaging, Sensors, Algorithms, and Applications, pp. 149-187, 2013

[5] I. Sanchez Ruiz, "Object tracking using direct methods in RGB-D cameras," Thesis (Master thesis), ETS de Ingenieros Informaticos, 2015

[6] M. Camplani, S. Hannuna, M. Mirmehdi, D. Damen, A. Paiement, L. Tao, and T. Burghardt, "Real-time RGB-D tracking with depth scaling kernelised correlation filters and occlusion handling," British Machine Vision Conference, 2015

[7] D. Greenhill, J. Renno, J. Orwell, and G.A. Jones, "Occlusion analysis: learning and utilising depth maps in object tracking," Image and Vision Computing, vol. 26, no. 3, pp. 430-441, 2008

[8] Y. Chen, Y. Shen, X. Liu, and B. Zhong, "3D object tracking via image sets and depth-based occlusion detection," J. Signal Processing, vol. 112, pp. 146-153, July 2015

[9] J. Lee, P. Karasev, and A. Tannenbaum, "Range based object tracking and segmentation," IEEE Conf. on Image Processing, pp. 4641-4644, 2010

[10] S. Ottonelli, P. Spagnolo, P.L. Mazzeo, and M. Leo, "Improved video segmentation with color and depth using a stereo camera," IEEE Int. Conf. on Industrial Technology, pp. 1134-1139, 2013

[11] J. Lee, S. Lankton, and A. Tannenbaum, "Object tracking and target reacquisition based on 3-D range data for moving vehicles," IEEE Trans. Image Processing, vol. 20 no. 10, pp. 2912-2924, 2011

[12] S.-C. Shen, W.-L. Zheng, and B.-L. Lu, "Online object tracking based on depth image with sparse coding," Neural Information Processing, pp. 234-231, 2014

[13] L. Jia, and R.J. Radke, "Using time-of-flight measurements for privacy-preserving tracking in a smart room," IEEE Trans. on Industrial Informatics, vol. 10, no. 1, pp. 689-696, 2013.

[14] H. Sabirin, H. Sankoh, and S. Naito, "Utilizing attributed graph representation in object detection and tracking for indoor range sensor surveillance cameras," IEICE Transactions on information and Systems, vol. 98, no. 12, pp. 2299-2307, 2015.

[15] T. Bagautdinov, F. Fleuret, and P. Fua, "Probability occupancy maps for occluded depth images," IEEE Conf. on Comp. Vision and Patt. Rec., pp. 2829-2837, 2015.

ISBN 978-0-9928626-7-1 © EURASIP 2017