Collaborative Multi-Sensor Image Transmission and Data Fusion in Mobile Visual Sensor Networks Equipped with RGB-D Cameras

Xiaoqin Wang, Ahmet Sekercioglu, Tom Drummond, Enrico Natalizio, Isabelle Fantoni, Vincent Fremont

Collaborative Multi-Sensor Image Transmission and Data Fusion in Mobile Visual Sensor Networks Equipped with RGB-D Cameras

Xiaoqin Wang[1] Y. Ahmet Şekercioğlu[2] Tom Drummond[1]
Enrico Natalizio[2] Isabelle Fantoni[2] Vincent Frémont[2]

***Abstract*— We present a scheme for multi-sensor data fusion applications, called *Relative Pose based Redundancy Removal (RPRR)*, that efficiently enhances the wireless channel utilization in bandwidth-constrained operational scenarios for RGB-D camera equipped visual sensor networks. Pairs of nodes cooperatively determine their own relative pose, and by using this knowledge they identify the correlated data related to the common regions of the captured color and depth images. Then, they only transmit the non-redundant information present in these images. As an additional benefit, the scheme also extends the battery life through reduced number of packet transmissions.**

Experimental results confirm that significant gains in terms of wireless channel utilization and energy consumption would be achieved when the RPRR scheme is used in visual sensor network operations.

I. Introduction

Visual sensor networks (VSNs) [1] allow the capture, processing and transmission of per-pixel color information from a variety of viewpoints. Low-cost RGB-D sensors, such as Microsoft Kinect [2], also add depth data to the collected information, and have attracted the interest of research community as a new way of capturing real-world scenes. The inclusion of RGB-D sensors makes VSNs to be capable of collecting color and depth data in cost-effective ways, and can significantly enhance the performance of applications such as immersive telepresence or mapping [3], environment surveillance [4], or object recognition and tracking [5] as well as opening the possibilities for new and innovative applications [6]. The value of VSN applications becomes even more important especially in places inaccessible to humans such as search and rescue operations after earthquakes or nuclear accidents. An illustrative scenario is shown in Fig. 1.

However, RGB-D sensors inevitably generate vast amounts of visual and depth data. The volume of data will be even larger when multiple camera sensors observe the same scene from different viewpoints and exchange/gather their measurements to obtain a better understanding of the environment. As a result, collected images will inevitably contain a significant amount of correlated and redundant information. Transmission of all the captured data will lead to unnecessarily high transmission load. As the sensors will most likely be communicating in ad hoc networking configurations, communication bandwidth will be at a premium [9], error-prone and not suitable for frequent data delivery in large quantities. Moreover, wireless transceivers consume a significant part of the available battery power [10], and limited capacity of on-board power sources should also be considered. Consequently, transmission of visual and depth information in resource-constrained VSN nodes must be carefully controlled and minimized as much as possible.

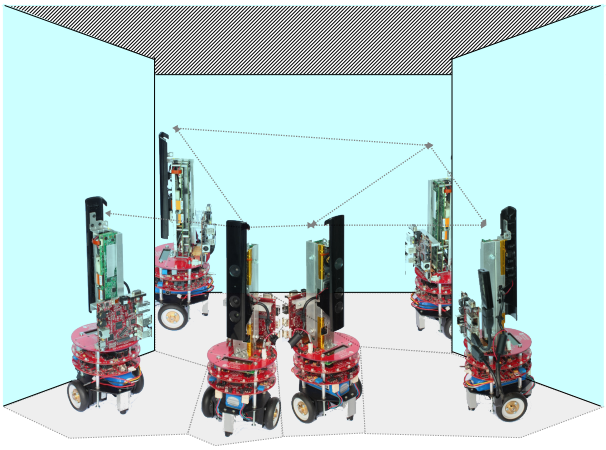

Fig. 1: An indoor mapping and exploration scenario showing the Monash University's RGB-D sensor equipped experimental mobile robots "eyeBugs" [7], [8]. The robots form a mobile ad hoc network and exchange information for performing various tasks.

In the next section we discuss the leading approaches that have been proposed to address this problem. Then, we present a brief overview of the RPRR framework in Section III. Detailed description of each stage can be found in Section IV. Experimental results are presented and analyzed in Section V, followed by our concluding remarks.

II. Related Works

A number of solutions can be found in the research literature that intend to remove or minimize the correlated

This work was supported by the Australian Research Council Centre of Excellence for Robotic Vision (project number CE140100016).

This work was carried out in the framework of the Labex MS2T, which was funded by the French Government, through the program "Investments for the Future" managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02).

[1]X. Wang and T. Drummond are with the ARC Centre of Excellence for Robotic Vision, Monash University, Victoria, 3800, Australia (e-mail: xiaoqin.wang@monash.edu; tom.drummond@monash.edu).

[2]Y. A. Şekercioğlu, E. Natalizio, I. Fantoni and V. Frémont are with the Sorbonne Universités, Université de Technologie de Compiègne, CNRS, UMR 7253 Heudiasyc, 60200 Compiègne, France (e-mail: asekerci@ieee.org; enrico.natalizio@utc.fr; isabelle.fantoni@utc.fr; vincent.fremont@utc.fr).

data for transmission in VSNs. They can be broadly classified into three groups:

1) optimal camera selection,
2) collaborative compression and transmission, and
3) distributed source coding.

The optimal camera selection algorithms [11], [12], [13] attempt to group the camera sensors with overlapping fields-of-view (FoVs) into clusters and only activate the sensor which can capture the image with the highest number of feature points. These algorithms operate under the assumption that the images captured by a small number of camera sensors in one cluster are good enough to represent the information of the scene/object. However, the occlusions in FoVs may cause significant differences between the images captured by cameras with very similar sensing directions. Therefore, the assumption is not realistic and this kind of approach is not applicable in many situations.

The collaborative compression and transmission methods [14], [15], [16] jointly encode the captured multi-view images. Only the uncorrelated visual content is delivered in the network after being jointly encoded by some recent coding techniques (for example, Multiview Video Coding (MVC) [17]). However, at least one node in the network is required to have the full set of images captured by the other sensors in order to perform image registration. Therefore, redundant information cannot be removed completely and still needs to be transmitted at least once. Moreover, as color images do not contain the full 3-D representation of a scene, these methods introduce distortions and errors when the relative poses (location and orientation) between sensors are not pure rotation or translation, or the scenes have complex geometrical structures and occlusions.

Distributed Source Coding (DSC) algorithms [18], [19], [20] are another group of promising approaches that can be used to reduce the redundant data in multiview VSN scenarios. Each DSC encoder operates independently, but at the same time, relies on joint decoding operations at the sink (remote monitoring station). However, the side information must be predicted as accurately as possible. The correlation structure can hardly be identified at the decoder side without an accurate knowledge of the network topology and the poses of the sensors. These are the main disadvantages that prevent DSC algorithms from being widely implemented. A detailed discussion on multi-view image compression and transmission schemes in VSNs is presented in [21].

The algorithms mentioned above focus only on color (RGB) information. Only a very limited number of studies have been reported [22], [23] so far that use RGB-D sensors in VSNs, as their use in VSNs has not yet become widespread. Our extensive review of the research literature has identified that no earlier studies have been published that attempt to develop an efficient coding system considering both color and depth information for optimizing the bandwidth and energy usage for wireless communications in VSNs equipped with RGB-D sensors. In this paper we focus on this issue, and present a novel approach to the development of a comprehensive solution for minimizing the transmission of redundant RGB-D data in VSNs. Our framework, called *Relative Pose based Redundancy Removal (RPRR)*, efficiently removes the redundant information captured by each sensor before transmission. We designed the RPRR framework particularly for RGB-D sensor equipped VSNs which anticipate that they will work in situations with severely limited communication bandwidth as support systems for disaster management or rescue and recovery operations.

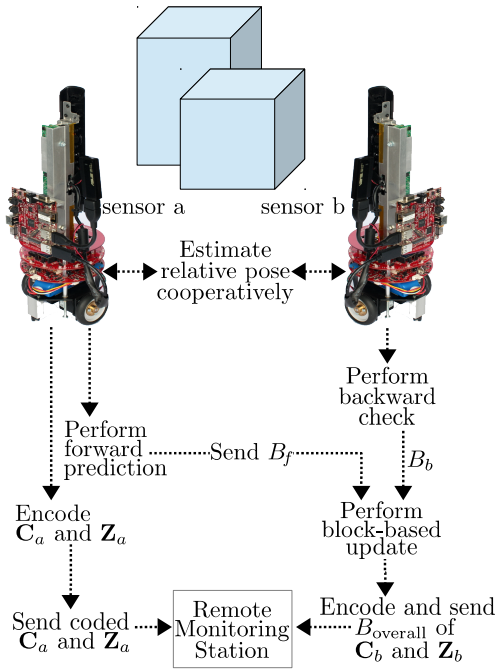

Fig. 2: Operational overview of the RPRR framework: (i) sensors cooperatively estimate their relative pose (ii) identify redundant sections of the scenery through forward prediction and backward check operations (iii) perform image coding and data transmission, and finally (iv) data fusion and image reconstruction is done at the remote monitoring station. Details of each stage can be found in Section IV.

In the RPRR framework, the characteristics of depth images, captured simultaneously with color data, are used to achieve the desired efficiency. Instead of using a centralized image registration technique [24], which requires one node to have full knowledge of the images captured by the others to determine the correlations, we propose a new approach based on relative pose estimation between pairs of RGB-D sensors [25] and 3-D image warping technique [26]. The method we propose locally determines the color and depth information, which can only be seen by one sensor but not the others. Consequently, each sensor is required to transmit only the uncorrelated information to the remote monitoring station.

III. System Overview

Here, we first provide a brief overview of the hardware and software components of the mobile RGB-D sensors we use, then present a summary of the operation of the framework

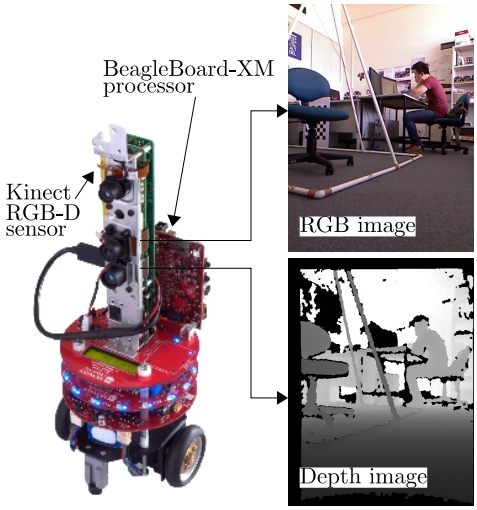

Fig. 3: eyeBug [7], [8], the mobile RGB-D sensor we use in our experiments. The color and depth data generated by the Kinect sensor is processed on a BeagleBoard-xM [27] computer running the GNU/Linux operating system.

as shown in Fig. 2. Details of each subsystem are presented in Section IV.

A. eyeBug: A Mobile RGB-D Sensor

The RPRR framework was implemented and tested by using the data captured by an experimental VSN platform developed in Monash University's Wireless Sensor and Robot Networks Laboratory (WSRNLab) [28]. The platform consists of multiple mobile RGB-D sensors named "eyeBug" (Fig. 3). EyeBugs were created for computer vision and robotics research, such as multi-robot SLAM or scene reconstruction. We selected the Microsoft Kinect as the RGB-D sensor, due to its low cost and wide availability. We placed a Kinect vertically at the center of the top board of each eyeBug. The value of each depth pixel represents the distance information in millimeters. Invalid depth pixel values are recorded as zero, indicating that the RGB-D sensor could not estimate the depth information of that point in the 3-D world. A BeagleBoard-xM single-board computer [27] is used for image processing tasks. A USB WiFi adapter is connected to the BeagleBoard to provide communication between robots.

B. Relative Pose Based Redundancy Removal (RPRR) Framework

In a mobile VSN tasked with mapping a region using RGB-D sensors, it is highly possible that multiple sensors will observe the same scene from different viewpoints. Consequently, scenery captured by the sensors with overlapping FoVs will have a significant level of correlated information. Here, our goal is to efficiently extract and encode the uncorrelated RGB-D information, and avoid transmitting the same surface geometry and color information repeatedly.

Consider the two sensors, a and b, of this VSN with overlapping FoVs. Let $\mathbf{Z}_a$ and $\mathbf{Z}_b$ denote a pair of depth images returned by sensors, and $\mathbf{C}_a$ and $\mathbf{C}_b$ are the corresponding color images. In the encoding procedure, we first estimate the location and orientation of one sensor relative to the other. In the second step, before encoding the depth and color images into a bitstream, the disparities between the RGB-D information captured by the two sensors are determined. To achieve this, forward prediction/backward check and block-based update using the relative pose information are performed to generate a prediction of $\mathbf{Z}_b$ in sensor a and to determine the depth information which only exists in $\mathbf{Z}_b$ but not in $\mathbf{Z}_a$. Then, only the uncorrelated information in $\mathbf{Z}_b$ is encoded and transmitted. As both the color and depth images are registered, only the color information in $\mathbf{C}_b$ corresponding to the uncorrelated depth information needs to be transmitted. Therefore, the redundancy in the RGB-D information is removed in the encoding process. A high-level overview of the process flow is shown in Fig. 2.

The information carried in the received bitstream is decoded and recovered. Then, to deal with the under-sampling issue [29], and to enhance the quality of the reconstructed color and depth images, we propose a number of post-processing approaches. A detailed explanation of each step is provided in the next section.

IV. Implementation Details of the RPRR Framework

In this section, we present the major functional blocks and algorithms of the RPRR framework: Relative pose estimation, forward/backward prediction, block-based update, the lossless differential coding scheme, and post-processing operations.

A. Relative Pose Estimation

The relative pose between RGB-D sensors a and b can be represented by a transformation matrix, $\mathbf{M}_{ab}$ in SE(3),

$$\mathbf{M}_{ab} = \begin{bmatrix} & \mathbf{R} & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

where $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t}$ is a 3×1 translation vector.

The ICP-BD algorithm, published in one of our earlier papers [25], determines the relative pose at a consistent real world scale through explicit registration of surface geometries extracted from two depth images. The registration problem is solved iteratively by minimizing a cost function, in which error metrics are defined based on the bidirectional point-to-plane geometrical relationship.

The cooperating beam-based sensor model [30] with ICP algorithm reduces the adverse effects in point cloud matching for situations where two views of a scene are partially seen by the sensors. Moreover, this algorithm only requires sensors to exchange a very small amount of depth information, which makes it bandwidth efficient, and so fits the requirements of VSNs.

B. Forward Prediction/Backward Check and Block-based Update

1) Forward Prediction: Let $\mathbf{p}_e = [x; y; z; 1]^T$ denote a real world point in Euclidean space. Given the intrinsic

parameters of the RGB-D sensor, $\mathbf{p}_e$ can be directly derived from the corresponding pixel in depth image as

$$\mathbf{p}_e \equiv \frac{1}{z}\begin{bmatrix} x & y & z & 1 \end{bmatrix} \equiv \begin{bmatrix} \frac{i-i_c}{f_x} & \frac{j-j_c}{f_y} & 1 & \frac{1}{z} \end{bmatrix}^T, \quad (2)$$

where (i, j) denotes the pixel coordinates in the depth image, (i_c,j_c) is the principal point and (f_x,f_y) is the focal length of the camera. We define that $\mathbf{p_e}$ can be observed by both homogeneous mobile RGB-D sensor a and b. The projections of $\mathbf{p}_e$ are located at pixel coordinates (i_a, j_a) and (i_b, j_b) on the depth images $\mathbf{Z}_a$ and $\mathbf{Z}_b$, respectively. Under the assumption that the world coordinates system is equal to the mobile sensor coordinate system, the depth pixel (projection) at (i_a, j_a) in $\mathbf{Z}_a$ can establish a relationship between the depth pixel at (i_b, j_b) in $\mathbf{Z}_b$ as follows,

$$\begin{bmatrix} \frac{i_b-i_c}{f_x} & \frac{j_b-j_c}{f_y} & 1 & \frac{1}{z_b} \end{bmatrix}^T = \mathbf{M}_{ab} \begin{bmatrix} \frac{i_a-i_c}{f_x} & \frac{j_a-j_c}{f_y} & 1 & \frac{1}{z_a} \end{bmatrix}^T \quad (3)$$

Therefore, with the accurate relative pose information $\mathbf{M}_{ab}$, sensor a can predict a depth image $\mathbf{Z}_b^*$, which is virtually captured at sensor b's viewpoint, by applying Eq. 3 on each pixel in $\mathbf{Z}_a$.

In this process, it can happen that two or more different depth pixels are warped to the same pixel coordinate in $\mathbf{Z}_b^*$. This over-sampling issue happens because some 3-D world points are occluded by the other ones at the new viewpoint. In order to solve this problem, we always compare the depth values of the pixels warped to the same coordinate. The pixel with the closest range information to the camera always overwrites the other pixels. As the depth image is registered to the color image, the color pixels in $\mathbf{C}_a$ can also be mapped along with the depth pixels to generate a virtual color image $\mathbf{C}_b^*$.

Then all of the captured images and virtual images are decomposed into 8×8 macro blocks. In the virtual depth image, some blocks have no depth information. This is because none of the pixels in $\mathbf{Z}_a$ can be warped to these regions. It indicates the blocks with the same coordinates in $\mathbf{Z}_b$ and $\mathbf{C}_b$ contain the information that can only be observed by sensor b while it cannot be seen by sensor a. Therefore, after sensor a transmits these block coordinates to sensor b, sensor b will record these block coordinates as a set, B_f, and only needs to transmit the RGB-D information in these blocks of $\mathbf{Z}_b$ and $\mathbf{C}_b$ to the remote monitoring station.

An example of this process is shown in Fig. 4. In this example, the regions containing the depth information that can only be observed by $\mathbf{Z}_b$ are outlined in yellow.

2) Backward Check and Block-Based Update: Although the forward prediction can detect the uncorrelated information in the images captured by the other sensor in most circumstances, it may fail to operate correctly in situations when some points are occluded by the objects that can only be seen by sensor b, but which cannot be observed by sensor a. A typical scenario is shown in Fig. 5. In this case, as the cylinder cannot be observed by sensor a, it will incorrectly treat the background (shown as the dashed rectangle) as the surface that can be observed by

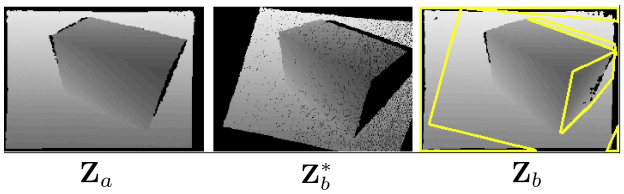

Fig. 4: An intuitive example of forward prediction. The depth image $\mathbf{Z}_b^*$ is predicted from $\mathbf{Z}_a$ as the image captured by sensor b virtually. The uncorrelated information in $\mathbf{Z}_b$ is outlined in yellow.

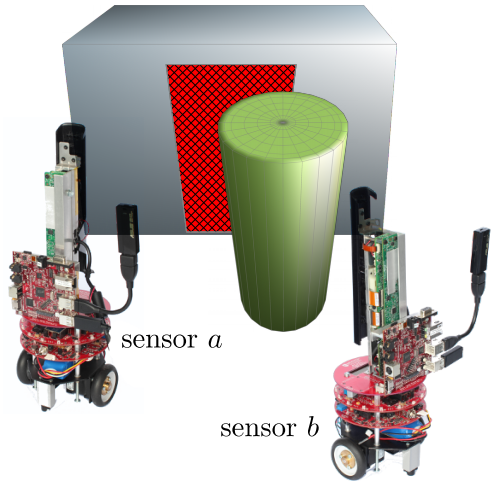

Fig. 5: The rectangular surface area in the background in the FoV of sensor b is occluded by the cylinder in the foreground.

sensor b. However, the surface of the cylinder is included in $\mathbf{Z}_b$, which occludes the background from the viewpoint of sensor b. Therefore, the forward prediction cannot accurately determine the uncorrelated depth and color information in such a situation.

In order to resolve this problem, we introduce a backward check mechanism. Similar to the warping process from sensor a to b, in the backward check process, sensor b can also generate virtual images $\mathbf{Z}_a^*$ and $\mathbf{C}_a^*$, which are virtually captured at sensor a's viewpoint. The warping process in the backward check can be described as

$$\begin{bmatrix} \frac{i_a-i_c}{f_x} & \frac{j_a-j_c}{f_y} & 1 & \frac{1}{z_a} \end{bmatrix}^T = \mathbf{M}_{ab}^{-1} \begin{bmatrix} \frac{i_b-i_c}{f_x} & \frac{j_b-j_c}{f_y} & 1 & \frac{1}{z_b} \end{bmatrix}^T. \quad (4)$$

Pixels at (i_b, j_b) in $\mathbf{Z}_b$ can be mapped to (i_a, j_a) in $\mathbf{Z}_a^*$. In this process, the pixels representing the range information of the surface of the cylinder will move out of the image coordinate range and will not be shown in $\mathbf{Z}_a^*$. Sensor b needs to transmit the image blocks which contain the information of the cylinder surface which cannot be seen by sensor a. Therefore, sensor b requires to determine the blocks including pixels in $\mathbf{Z}_b$ that move out of the image range in the backward check process. The set of these block coordinates is B_b. Then, sensor b will derive the universe of the block coordinate sets B_f and B_b as $B_{\text{overall}} = B_f \cup B_b$.

The blocks of B_{overall} in $\mathbf{Z}_b$ and $\mathbf{C}_b$ contain the information which can only be observed by sensor b.

Therefore, each sensor can easily determine the uncorrelated RGB-D information by using only the relative pose information, and consequently avoid transmitting/receiving and comparing complete color and depth images: sensor a sends the complete captured color and depth images, while sensor b sends only the information in B_{overall} to the remote monitoring station. As we show later in the paper, this leads to significant bandwidth saving.

C. Post-Processing at Decoder Side

After the removal of the redundant information, the uncorrelated color/depth information is compressed to improve the efficiency of the communication channel usage. For this purpose, the depth data is encoded by entropy coding scheme. For RGB color data, we use the Progressive Graphics File (PGF) scheme presented in [31].

At the decoder side, the received bitstream is decoded with the same look-up tables used at the encoder side. After the color and depth images captured by sensor a have been decoded, we use these decoded images to predict the depth and color images captured by sensor b.

The 3-D image warping process (Eq. 3) may introduce some visual artifacts in the synthesized view, such as cracks and ghosting artifacts [32]. Cracks are small disocclusions, and mostly occur due to under-sampling. Ghosts are artifacts due to the projection of pixels that have background depth and mixed foreground/background color. Various methods [33], [34] have been proposed in the literature to prevent these artifacts. We adopt the recovery scheme proposed in [35] and an adaptive median filter to remove the cracks and ghosts artifacts respectively.

V. Experimental Results and Performance Evaluation

In this set of experiments, we evaluated the performance of the RPRR framework by using two mobile RGB-D sensors of our VSN platform. Color and depth images were captured in four different scenes, as shown in Fig. 6. In this set-up, sensor a transmits entire captured color and depth images to a central station (receiver). Then, sensor b is required to transmit only the uncorrelated color and depth information that cannot be observed by sensor a. At the receiver, the color and depth images captured by sensor b are reconstructed using the information transmitted by two sensors. As the entire color and depth images captured by sensor a are compressed and transmitted to the receiver, we only had to evaluate the reconstruction quality of the images captured by sensor b. The depth images are usually complementary to the color images in many applications, and in our framework the color images are reconstructed according to depth image warping. The reconstructed color images are necessarily related to the reconstructed depth images. If the color images can be accurately reconstructed, the reconstructed depth images are also precise. Therefore, in this set of experiments we focused on evaluating the quality of the reconstructed color images.

A. Subjective Evaluation

The image blocks transmitted by sensor b are shown in the third row of Fig. 6. The fourth row of the figure illustrates the images reconstructed by using them.

It can be seen that the images captured by sensor a have been warped and stitched to generate the reconstructed color images captured by sensor b. In the reconstructed images of scene 2 and 4, we also observe significant color changes over the stitching boundary. This is because the illumination is inconsistent in the scene and the images captured by the sensors have different levels of brightness. Generally, it is clear that the reconstructed images preserve the structural information of the original images accurately.

B. Objective Evaluation

Even though many approaches have been proposed to compress multi-view images, they cannot be applied to our system. These approaches either require the transmitter to have knowledge of the full set of images or only work on cameras with very small motion differences. In contrast, in our case, each sensor only has its own captured image, and the motion difference between two visual sensors is very large. To the best of our knowledge, this is the first distributed framework to efficiently code and transmit images captured by multiple RGB-D sensors with large pose differences, and so, we do not have any work to compare ours against. For this reason, we can only compare the performance of our framework with the approaches which compress and transmit images independently.

By adjusting the compression ratio of the coding scheme, RPRR framework can vary its coding performance. The performance was evaluated according to the following two aspects: reconstruction quality and compression ratio. We measured the Peak-Signal-to-Noise-Ratio (PSNR) between the reconstructed and original images captured by sensor b with different compression ratios. The results are shown in Fig. 7.

Fig.7 shows that the RPRR framework can achieve much higher compression ratios than the independent transmission scheme. When the image quality of the reconstructed images are the same (as measured by PSNR values), the average compression ratio achieved by the RPRR framework is 174.6% higher than the independent transmission scheme. However, it should be noted that, the PSNR upper bounds achieved by RPRR framework have limits. It is because the reconstruction quality depends on the depth image accuracy and correlations between color images. Since the depth images generated by a Kinect sensor is not accurate enough, the displacement distortion of depth images, especially the misalignment around the object edges, introduces noise in the reconstruction process. Another reason is the inconsistent illumination between the color images captured by two sensors. Even if the forward prediction/backward check process establishes the correct correspondences between two color pixels according to the transformation between depth images, the values of these two color pixels can be very different due to the various brightness levels in two images.

(a) Scene 1 (b) Scene 2 (c) Scene 3 (d) Scene 4

(e) Scene 1 (f) Scene 2 (g) Scene 3 (h) Scene 4

(i) Scene 1 (j) Scene 2 (k) Scene 3 (l) Scene 4

(m) Scene 1 (n) Scene 2 (o) Scene 3 (p) Scene 4

Fig. 6: A demonstration of the scheme over four sets of images: First and second rows show the images captured by sensors a, and b respectively. In the third row, image blocks transmitted by sensor b are shown (here black regions denote the image blocks that are not transmitted). The fourth row shows the reconstructed images at the receiver side using the data sent by sensor b.

Although the structures of the scenes are preserved nicely in the reconstructed color images, distinct color changes over the stitching boundaries are shown in Fig. 6 (n) and (p). This is the reason that the reconstructed images in Scene 3 have the highest PSNR while the reconstructed images in Scene 2 have the lowest PSNR. These characteristics lead to low PSNR upper bounds of the reconstructed color images. A number of methods [36], [37] have been proposed to overcome this drawback, however the time-complexity of these methods prevents them from being implemented on computationally-constrained sensors. Consequently, we can say that the RPRR framework is suitable for implementation of the applications with very limited bandwidth which require very high compression ratios. This is because when the compression ratio increases, the quality of the color image reconstructed by RPRR decreases more slowly than the quality of the image compressed by the independent transmission scheme. Due to the large amount of captured color/depth data and limited bandwidth, our proposed RPRR framework suits the needs of VSNs equipped with RGB-D sensors.

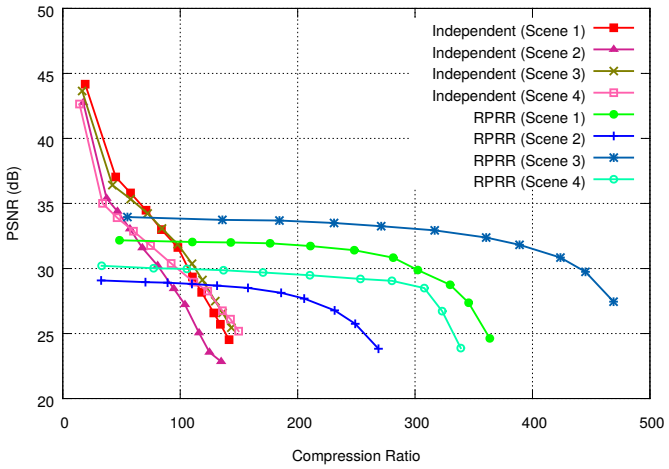

Fig. 7: Comparisons of PSNR (dB) achieved by compressing the images at various levels by using the RPRR framework against transmitting them independently.

C. Evaluation of Energy Consumption and Amount of Transmitted Data

Also, the limited battery capacity of mobile sensors places limits on their performance, a data transmission scheme while attempting to reduce the transmission load, must not have a significant negative impact on the overall energy consumption. In this section, we present our experimental measurements and evaluation regarding the overall energy consumption and amount of transmitted data of the RPRR framework collected on our eyeBug mobile visual sensors to demonstrate this aspect.

The overall energy consumption of the RPRR framework can be measured by

$$\begin{aligned} E^{R}_{\text{overall}} &= E_{\text{processing}} + E_{\text{encoding}} + E_{\text{sending}} \\ &= V_o I_p t_p + V_o I_e t_e + V_o I_s t_s \end{aligned} \tag{5}$$

in which V_o denotes the sensor's operating voltage, and I_p, I_e, and I_s represent the current drawn from the battery during processing, encoding, and sending operations. t_p, t_e, and t_s are the corresponding operation times required for these procedures.

The overall energy consumption when images are transmitted independently can be measured as,

$$\begin{aligned} E^{I}_{\text{overall}} &= E_{\text{encoding}} + E_{\text{sending}} \\ &= V_o I_e t_e + V_o I_s t_s. \end{aligned} \tag{6}$$

Note that, the operation times t_e and t_s are different in the two transmission schemes as the image sizes change after removing the redundant information.

Our sensor operates at 15 V, and the current levels remain fairly constant during each operation. We measured them as follows: $I_p = 0.06$ A, $I_e = 0.06$ A, and $I_s = 0.12$ A. Our experiments show that in the RPRR framework, due to different compression ratios, the transmission time varies between 32 and 42 ms, and the operational time for processing and encoding remains between 509 between 553 ms. The overall energy consumption of the RPRR scheme changes between 480 and 520 mJ, depending on the compression ratio. The corresponding values for the independent scheme are between 918 and 920 mJ. The data clearly show that the RPRR framework leads to the consumption of much lower battery capacity than the independent transmission scheme. It cuts the overall energy consumption of the sensor nearly by half. In the RPRR framework, the energy consumption for two sensors are asymmetric, and if sensor a always transmits complete images, its energy will be quickly drained. A simple method to prolong the network lifetime is for the two sensors to transmit complete images alternately. The current consumed by an eyeBug in idle status is 650 mA. According to the experimental results above, the theoretical operational time of RPRR on two eyeBugs with 2500 mAh 3-cell (11.1 V) LiPo batteries is around 5.2 hours. In this period, around 32400 color and depth image pairs can be transmitted to the remote monitoring station.

VI. Discussion and Concluding Remarks

In this paper, we introduced a multi-sensor data fusion framework that efficiently removes the redundant visual information captured by the RGB-D sensors of a mobile VSN. In our work, we considered a multiview scenario in which pairs of sensors observe the same scene from different viewpoints. By taking advantage of the unique opportunities offered by depth images, our scheme identifies the correlated regions between the images captured by these sensors using only the relative pose information. Then, only the information related to the uncorrelated regions is transmitted. This approach significantly reduces the amount of information transmitted compared with sending two individual images independently. In addition, through our experimental platform we demonstrated that the scheme's computational resource requirements are quite modest, and it can run on battery-operated sensor nodes. The experimental results confirm that the compression ratio achieved by the RPRR framework is nearly twice the independent transmission scheme, and it accomplishes this result while almost halving the energy consumption of the independent transmission scheme on average.

The RPRR framework is the first attempt to remove the redundancy in the color and depth information observed by VSNs equipped with RGB-D sensors. Our scheme, however, operates only on pairs of mobile sensors at this stage. A relatively straightforward extension of the RPRR framework for networks with large number of RGB-D sensors could be by selecting one sensor as "the reference" one that transmits complete images (like sensor a in Fig. 2) while the other sensors transmit only the uncorrelated information (like sensor b in Fig. 2). However, this approach will not be sufficient to eliminate all the redundant information and further refinements are possible. In the next stage of our research efforts we will concentrate on developing a more sophisticated method which will use feature matching algorithms to assign sensors with overlapping FoVs to the same subgroups to apply RPRR on sensors in the same subgroup.

We expect that this method will remove redundancies very effectively in VSNs consisting of a large number of RGB-D sensors.

REFERENCES

[1] S. Soro and W. Heinzelman, "A Survey of Visual Sensor Networks," *Advances in Multimedia*, vol. 2009, 2009.

[2] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced Computer Vision With Microsoft Kinect Sensor: A Review," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1318–1334, 2013.

[3] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *Proceedings of the International Conference on Intelligent Robot Systems (IROS 2012)*, Vilamoura, Portugal, Oct. 2012, pp. 573 – 580.

[4] W. Liu, T. Xia, J. Wan, Y. Zhang, and J. Li, *RGB-D Based Multi-attribute People Search in Intelligent Visual Surveillance*, 2012, vol. 7131, pp. 750–760.

[5] D. Alexiadis, D. Zarpalas, and P. Daras, "Real-Time, Full 3-D Reconstruction of Moving Foreground Objects From Multiple Consumer Depth Cameras," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 339–358, Feb 2013.

[6] C. Wang, Z. Liu, and S.-C. Chan, "Superpixel-Based Hand Gesture Recognition With Kinect Depth Camera," *IEEE Transactions on Multimedia,*, vol. 17, no. 1, pp. 29–39, Jan 2015.

[7] N. D'Ademo, W. L. D. Lui, W. H. Li, Y. A. Şekercioğlu, and T. Drummond, "eBug: An Open Robotics Platform for Teaching and Research," in *Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2011)*, Melbourne, Australia, Dec. 2011.

[8] "eyeBug - a Simple, Modular and Cheap Open-Source Robot," http://www.robaid.com/robotics/eyebug-a-simple-and-modular-cheap-open-source-robot.htm, Sep. 2011.

[9] J. Li, C. Blake, D. S. De Couto, H. I. Lee, and R. Morris, "Capacity of Ad Hoc Wireless Networks," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom 01)*, 2001, pp. 61–69.

[10] A. A. Aziz, Y. A. Şekercioğlu, P. Fitzpatrick, and M. Ivanovich, "A Survey on Distributed Topology Control Techniques for Extending the Lifetime of Battery Powered Wireless Sensor Networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 121–144, 2013.

[11] Y. Bai and H. Qi, "Redundancy Removal Through Semantic Neighbor Selection in Visual Sensor Networks," in *Proceedings of the Third ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2009)*, 2009, pp. 1–8.

[12] ——, "Feature-Based Image Comparison for Semantic Neighbor Selection in Resource-Constrained Visual Sensor Networks," *EURASIP Journal on Image and Video Processing*, vol. 2010, no. 1, p. 469563, 2010.

[13] S. Colonnese, F. Cuomo, and T. Melodia, "An Empirical Model of Multiview Video Coding Efficiency for Wireless Multimedia Sensor Networks," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 1800–1814, Dec 2013.

[14] W. C. Chia, L.-M. Ang, and K. P. Seng, "Multiview Image Compression for Wireless Multimedia Sensor Network Using Image Stitching and SPIHT Coding with EZW Tree Structure," in *Proceedings of the International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC 2009)*, vol. 2, 2009, pp. 298–301.

[15] W. C. Chia, L. W. Chew, L.-M. Ang, and K. P. Seng, "Low Memory Image Stitching and Compression for WMSN Using Strip-based Processing," *International Journal on Sensor Networks*, vol. 11, no. 1, pp. 22–32, Jan 2012.

[16] S. Colonnese, F. Cuomo, and T. Melodia, "Leveraging Multiview Video Coding in clustered Multimedia Sensor networks," in *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 475–480.

[17] A. Vetro, T. Wiegand, and G. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 626–642, 2011.

[18] N. Deligiannis, F. Verbist, A. C. Iossifides, J. Slowack, R. V. de Walle, R. Schelkens, and A. Muntenau, "Wyner-Ziv Video Coding for Wireless Lightweight Multimedia Applications," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 1–20, 2012.

[19] D. Chen, D. Varodayan, M. Flierl, and B. Girod, "Wyner-Ziv Coding of Multiview Images with Unsupervised Learning of Disparity and Gray Code," in *Proceedings of 15th IEEE International Conference on Image Processing (ICIP)*, 2008, pp. 1112–1115.

[20] N. Gehrig and P.-L. Dragotti, "Geometry-Driven Distributed Compression of the Plenoptic Function: Performance Bounds and Constructive Algorithms," *IEEE Transactions on Image Processing,*, vol. 18, no. 3, pp. 457–470, 2009.

[21] X. Wang, Y. A. Şekercioğlu, and T. Drummond, "Multiview Image Compression and Transmission Techniques in Wireless Multimedia Sensor Networks: A Survey," in *Proceedings of the 7th ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2013)*, Palm Springs, USA, 2013.

[22] E. Almazan and G. Jones, "Tracking People Across Multiple Non-Overlapping RGB-D Sensors," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2013)*, Jun. 2013, pp. 831–837.

[23] J. Shen, P.-C. Su, S. Cheung, and J. Zhao, "Virtual Mirror Rendering With Stationary RGB-D Cameras and Stored 3-D Background," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp. 3433–3448, Sept 2013.

[24] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Efficient Prediction Structures for Multiview Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1461–1473, 2007.

[25] X. Wang, Y. A. Şekercioğlu, and T. Drummond, "A Real-Time Distributed Relative Pose Estimation Algorithm for RGB-D Camera Equipped Visual Sensor Networks," in *Proceedings of the 7th ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2013)*, Palm Springs, USA, 2013.

[26] C. Fehn, "Depth-Image-Based Rendering (DIBR), Compression, and Transmission for a New Approach on 3D-TV," in *Proceedings of SPIE*, vol. 5291, 2004, pp. 93–104.

[27] "Beagleboard-xM System Reference Manual," http://beagleboard.org/static/.

[28] "Wireless Sensor and Robot Networks Laboratory (WSRNLab)," http://wsrnlab.ecse.monash.edu.au.

[29] W. R. Mark, L. McMillan, and G. Bishop, "Post-Rendering 3D Warping," in *Proceedings of the Symposium on Interactive 3D Graphics*, Providence, USA, 1997, pp. 7–ff.

[30] M. Krainin, K. Konolige, and D. Fox, "Exploiting Segmentation for Robust 3D Object Matching," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 4399–4405.

[31] C. Stamm, "A New Progressive File Format for Lossy and Lossless Image Compression," in *Proceedings of International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision,*, Czech Republic, 2002, pp. 30–33.

[32] Y. Morvan, "Acquisition, Compression and Rendering of Depth and Texture for Multi-View Video," Ph.D. dissertation, Technische Universiteit Eindhoven, Netherlands, 2009.

[33] M. Xi, J. Xue, L. Wang, D. Li, and M. Zhang, "A Novel Method of Multi-view Virtual Image Synthesis for Auto-stereoscopic Display," in *Advanced Technology in Teaching*, ser. Advances in Intelligent and Soft Computing, 2013, vol. 163, pp. 865–873.

[34] G. P. Fickel, C. R. Jung, and B. Lee, "Multiview Image and Video Interpolation Using Weighted Vector Median Filters," in *Proceedings of 2014 IEEE International Conference on Image Processing (ICIP)*, Oct 2014, pp. 5387–5391.

[35] L. Do, S. Zinger, Y. Morvan, and P. de With, "Quality Improving Techniques in DIBR for Free-Viewpoint Video," in *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, May 2009, pp. 1–4.

[36] K. Vijayanagar, M. Loghman, and J. Kim, "Refinement of Depth Maps Generated by Low-Cost Depth Sensors," in *SoC Design Conference (ISOCC), 2012 International*, Nov. 2012, pp. 355–358.

[37] ——, "Real-Time Refinement of Kinect Depth Maps using Multi-Resolution Anisotropic Diffusion," *Mobile Networks and Applications*, vol. 19, no. 3, pp. 414–425, 2014.