4-2014

Eliminating Mutual Views in Fusion of Ranging and RGB-D Data From Robot Teams Operating in Confined Areas

Damian M. Lyons
Fordham University

Karma Shrestha
Fordham University

Eliminating Mutual Views in Fusion of Ranging and RGB-D Data From Robot Teams Operating in Confined Areas.

Damian M. Lyons, Karma Shrestha
Fordham University, Robotics and Computer Vision Laboratory, Bronx NY 10458

ABSTRACT

We address the problem of fusing laser and RGB-Data from multiple robots operating in close proximity to one another. By having a team of robots working together, a large area can be scanned quickly, or a smaller area scanned in greater detail. However, a key aspect of this problem is the elimination of the spurious readings due to the robots operating in close proximity. While there is an extensive literature on the mapping and localization aspect of this problem, our problem differs from the dynamic map problem in that it involves at one kind of transient map feature, robots viewing other robots, and we know that we wish to completely eliminate all such mutual views.

In prior work, we investigated the problem of fusing laser data from multiple robots in such a manner as to reject this spurious data from other robots. This work showed that a combination of local robot-based direction filtering and global map-based visibility filtering at a central map server removed 91% of the spurious data and resulted in a 98% quality improvement. In this paper we additionally consider the problem of fusing RGB-D data generated by a stereo-camera sensor. An approach based on a model of human visual attention is presented and compared with our prior work and with other related work. This approach is an order of magnitude faster than the prior work and yet rejects 73% of the spurious data producing a 55% quality improvement. Results are shown for this approach for two experiments with a two robot team operating in a confined indoor environment (4m x 4m).

Keywords: cognitive robotics, navigation, sensory fusion.

1. INTRODUCTION

A team of robots can be used to very quickly build a situational awareness model of a threat site such as a bomb, or other hazardous material, planted in a building or for an urban disaster area. We address the general problem of deploying a large team of robots, with each robot equipped with a different and diverse sent of sensors, and with the objective of constructing a 3D model of the threat site as quickly as possible. In fact, the objective is to be able to construct such a model within minutes of deployment and to have that model be available at a remote command center for the purpose of situational awareness and threat response planning.

Using teams of robot to construct maps is a common research topic [1] [2]. The main focus of attention in that work is the efficient fusion of the individual sensor scans into an accurate 2D or 3D building map. While this is an important problem, our time limitation constrains us to worry also about how to quickly build the map given the huge amounts of sensor data from each robot, how to carry out the team motion planning [3], the incremental integration of maps and sensory data as robots move in and out of contact with each other, and the incremental transmission of information from groups of robots to the command center. In particular in this paper, and previously in [4], we address the issue of the relatively large amount of *mutual views* - scan data collected by robot team members of other team members. We have shown that this mutual view data introduces error, and when removed, substantially improves the fitting of planes to the scan data.

One of the most studied areas in robotics research has been the use of sensory data from one or from multiple robots to construct a map of the area around the robot or robots. Starting with work in occupancy maps derived from sonar data, this field has progressed to registration of successive laser scans on a single robot [5], fusion of 2-D maps from multiple robots [6], and Simultaneous Localization and Mapping (SLAM) for multiple robots [1] [7]. Extending to 3D, [8] exploits parallelism to address the additional computational complexity of 3-D scan registration, whereas [9] accomplish the 3-D registration of two robots by using external fiducial markings.

While dealing with a dynamic environment is an ongoing research problem [10], approaches fall into two classes: either the data association stage employs clustering to separate dynamic from static objects [10], or objects are tracked separately from SLAM and removed [11]. While this latter method is effective, it requires the mechanisms of object tracking.

In prior work [4], we developed several approaches to filter the mutual view data. Although the positions of the robot team members are available, the simple approach of using this information to filter mutual views is not effective for several reasons. Because robots move in and out of ad-hoc wireless connection with each other, they do not always have up to date information on each other's location. Secondly, even in the case where robots are in direct communication, their motions are planned locally [3] and hence they move erratically with respect to one another. The combined filter developed in [4] used an uncertain robot direction measurement in conjunction with map frequency filtering to eliminate 91% of the mutual view data. However, that work just used laser datasets, which are much sparser than stereo-camera (or Kinect) datasets. This combined method is extremely slow for stereo-camera datasets. Instead here we propose an approach based on a model of human visual attention, the *Global Workspace Theory* (GWT) [12] [13], that is much faster while still removing a large portion of the mutual view data.

In the next section, the data collection architecture we have developed is presented, including the two different 3-D laser-scanning robot configurations and stereo-camera robot configuration used in the team. Section 3 describes the dataset and metrics used for the prior and current evaluations, and also the evaluation of estimated best performance using the floor plane metric. The filtering algorithms developed in [4] and their performance and timing are briefly reviewed as motivation. Section 4 presents the proposed visual attention based model and the results obtained by applying this to two stereo-camera datasets. Section 5 summarizes and comments on future work.

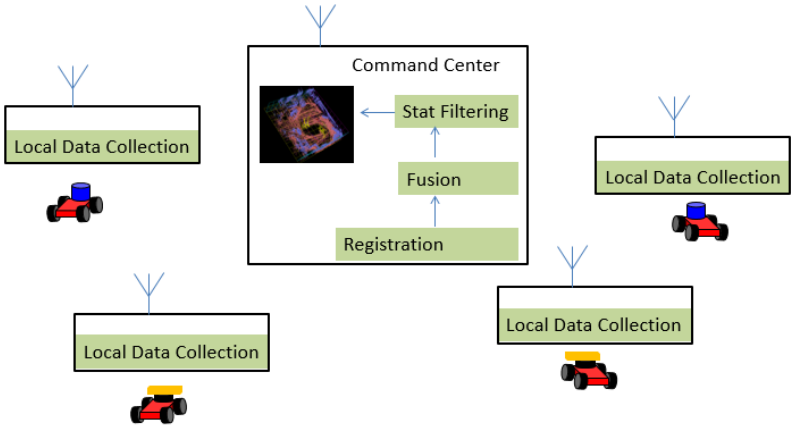

Figure 1: Robot Team Data Collection Architecture.

2. DATA COLLECTION ARCHITECTURE

In prior work [4] we have described the application problem we are address: We are specifically interested in surveillance, reconnaissance or search and rescue applications where a team of robots is deployed to a building or other confined area and must operate *quickly* to assembly a model of the area. We assume a heterogeneous, large team with the number and types of robot selected so as to most quickly 'swamp' the building and construct the map. We have constructed a straight-forward software architecture to investigate issues that arise in this kind of application, shown in Figure 1.

Data collected by each robot team member is locally processed, and then, along with the robot pose information for each team member, it is transmitted to the central map server. The map server time-stamps and stores all the data that is transmitted – in the Point Cloud Library (PCL) format – for later analysis or inspection. It registers data from each robot based on the accompanying pose information, which is Kalman-filtered to account for uncertainty in robot motion. The map server then fuses the information from each robot into a single point cloud and uses PCL to display the data collected. The filtering box shown in the server in Figure 1 refers to the filtering algorithms discussed here and in [4]. All the point cloud imagery shown in this paper were visualized using PCL.

Three classes of robot team member have been integrated into the architecture: two laser-based robots with the lasers mounted in different configurations and a stereo-camera based robot. Both laser-based robots and their laser configurations are shown in Figure 2. We use a simple modification of the common horizontal laser configuration to allow us to generate 3-D information as a robot moves: a single, tilted laser.

Figure 2(a) shows the down-laser robot, a *Pioneer 3-AT* which has its *SICK LMS-200* laser mounted tilted downwards. Figure 2(b) illustrates how when the robot moves, it scans the laser plane over the terrain ahead, including any objects. The limitation is that nothing higher than the center of the laser can be scanned. Figure 2(c) shows the up-laser robot,

which has its laser mounted tilted upwards. Figure 2(c) illustrates how when the robot moves forward the laser plane is scanned over parts of the scene above the center of the laser. These two robots are complementary in that one can map the 3-D structure below the height of the laser scanner, and one maps the structure above the laser scanner. Both operating together produces a full 3-D model. However, if both operate in close proximity, then they may scan each other. Figure 2(e) shows a Pioneer 3-AT with a *BumbleBee 2* stereo-camera mounted on a *Directed Perception* PT base. The robot generates both color and depth information (RGB-D) for any points in its stereo region (Figure 2(f)) as long as there is sufficient texture. This paper builds on [4] by adding two stereo-camera models to the up/down laser robot team.

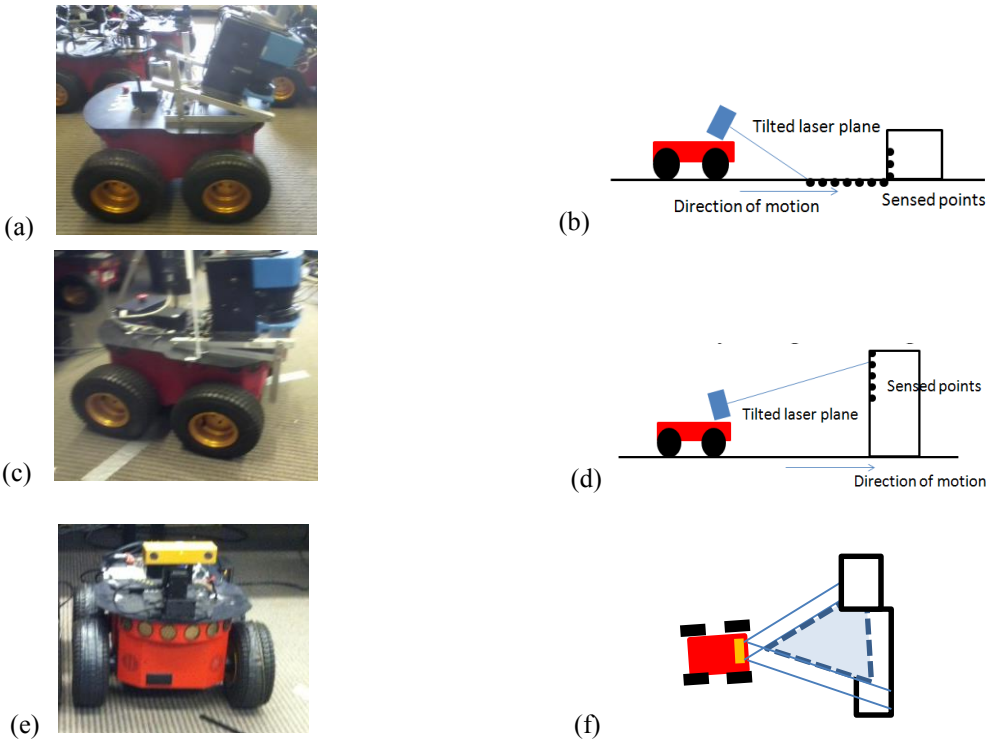

Figure 2: (a) Down laser mounted on Pioneer 3-AT; (b); Scanning action of down-laser; (c) Up laser mounted on Pioneer 3-AT; (d) Scanning action of up-laser; (e) Stereo-camera mounted on Pioneer 3-AT; (f) Stereo field of view

3. METRICS AND LASER FILTERS

Several combined datasets from a team of one up-laser and one down-laser robot have been collected. Figure 3 shows one of these, the *Lap2* dataset. The oval track shown in Figure 3(a) was tracked by both up-laser and down-laser robot, with one robot being approximately one half oval ahead of the other. Each robot completed two entire laps of the oval track. The resulting data set is displayed as a point cloud in Figure 3(b) from the top-down, and Figure 3(c) looking from the side. The data is color-coded (shaded) as follows: The up-laser data is all shown in a light shade of gray, whereas the down-laser data is all shown in a darker shade of gray. The points at which one robot scanned the second, which we will refer to for convenience as *ghost points*, are shown in black. Figure 3(d) shows the result of a two stereo-camera team following the same track. For clarity of comparison to the laser data, the stereo RGB data has all been mapped to gray, and the views of one robot by the other has been mapped to black. The stereo cloud is larger and denser than the laser cloud, but otherwise the mutual view problem is unchanged.

3.1 Performance Metrics

Several metrics have been introduced to measure the effect of the mutual view data. We focus on the accuracy of estimation of the floor plane as an example of the kind of processing that needs to happen to the point cloud data. Table 1 shows the results of this measurement for two laser datasets in their original form (first three columns) and with all the ghost data eliminated by hand (second three columns), from [4]. In each case, the first column shows the average error, the second

shows the percentage of all the points that were inliers on the plane and the third column shows the percentage of the inliers that were ghost points (and hence possibly corrupting the plane estimation). The final column shows that eliminating the mutual view data would produce a potential improvement in the average error of 98%. This observation motivates our development and evaluation of methods to automatically filter this mutual view data.

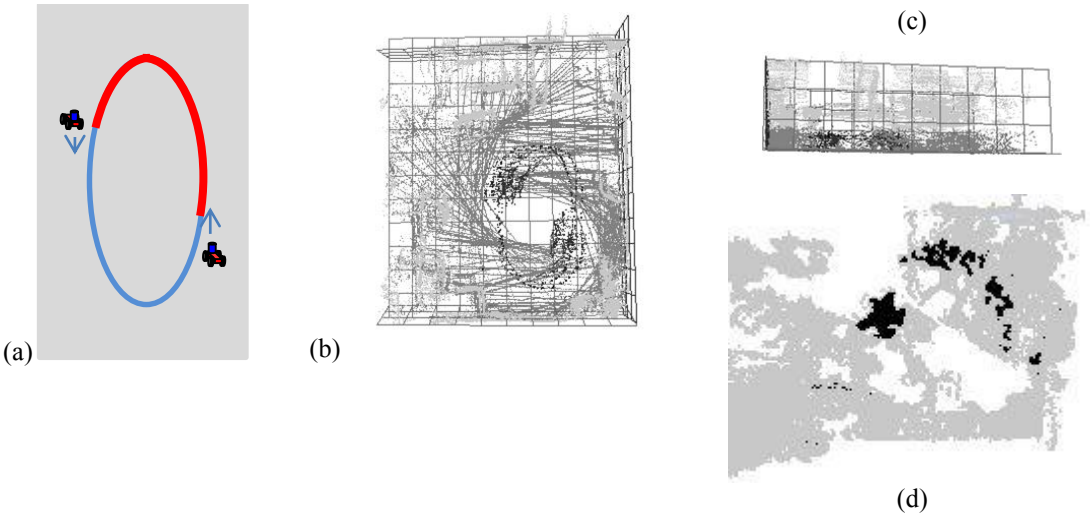

Figure 3: (a) Up-laser (dark) and down-laser (light) oval path (b, c); Color-coded data collected (down-laser, dark gray; up-laser, lighter gray; scans showing 'other' robot, black; each robot path is shown by superimposed shaded points. The floor and wall planes shown are estimated from the data; (c) shows a gray-level stereo-camera point cloud, looking top down, with the 'other' robot again shown in black.

Table 1: Plane Error for two datasets with, and without, mutual view data. The floor plane is fitted by RANSAC and represented in the form $ax+by+cz-d=0$. The average error is calculated by evaluating $\epsilon = ax+by+cz-d$ for all the RANSAC inliers on the floor plane and averaging the result.

	Including mutual view data			**Not including mutual view data**			
	Error (mm)	***% Inliers***	***% Ghosts***	***Error (mm)***	***% Inliers***	***% Ghosts***	***% Improve***
Lap	-3.1	82%	75%	0.054	83%	0%	98%
Fig8	-1.5	82%	55%	-0.046	83%	0%	97%

For each method for automatic removal of mutual data that we developed in [4] and which we will extend in the next section, we will measure effectiveness of the method with four metrics as follows:

1. % Ghost Points: This is the percentage of the mutual view data that was *identified* by the method. The larger this number, the more ghost points were identified and so the better.
2. % Other Points: Is the percentage of non-mutual view data that was identified by the method. The smaller this number the fewer non-mutual view points were mistakenly identified, and so the better.
3. % Improvement in Plane Error: This is the percentage improvement in the average error for the floor plane after filtering. The bigger this number, the smaller the error, and so the better.
4. % Improvement in Plane Inliers: This is the percentage improvement of the percentage of RANSAC inliers for the plane. The bigger this number, the more points are considered inliers for the plane, and so the better.

3.2 Laser-based Filtering Methods

Four mutual data filtering methods were investigated in [4]: two map-based methods, operating on the entire point cloud in the server, and two robot-based methods, operating on the data from each robot prior to registration and fusion in the server. To calculate the map spatial statistics, space is represented as a voxel array $V(x,y,z)$. The resolution r of the array is the dimension of a single voxel. In our examples, the entire space is a cube of $10\times10\times10$ *meters*.

Let $S(k,t)$ be the set of points in the laser scan of robot k at time t. For each $p \in S(k,t)$, $p = (x,y,z)$ in global map coordinates. The full point cloud that comprises the map is:

$$S = \bigcup_{k\epsilon\{R1,R2\}} \bigcup_{t\epsilon\{t0\cdots tn\}} S(k,t) \quad (1)$$

The voxel array is populated by the frequency of occurrence of point locations in the cloud:

$$V(b) = \sum_{p \in S} \delta_{pb} \qquad \delta_{pb} = \begin{cases} 1 & b = \left(\frac{x}{r}, \frac{y}{r}, \frac{z}{r}\right) \\ 0 & else \end{cases} \tag{2}$$

We can now define the *map-based frequency filtering algorithm* by the rule:

$$Filter(p)\ if:\ b = \left(\frac{x}{r}, \frac{y}{r}, \frac{z}{r}\right)\ and\ V(b) < \mu_f - k\sigma_f \tag{4}$$

Where the distribution $N(\mu_f, \sigma_f^2)$ is the frequency distribution calculated for *V(b)* and *k is* a constant.

The moving robots that give rise to the mutual view data points occlude floor data points, and hence we can define a *map-based visibility filtering algorithm*. If for any point we scan we find a voxel lower (smaller in Z) in the voxel array, then we should filter this point:

$$Filter(p)\ if:\ b = \left(\frac{x}{r}, \frac{y}{r}, \frac{z}{r}\right)\ and\ \exists b' = \left(\frac{x}{r}, \frac{y}{r}, k\right) \wedge \left(k < \frac{z}{r}\right) \tag{5}$$

Each robot only knows its own location with some degree of uncertainty based on its odometry and localization procedures; thus, if we use robot position as a mutual view filtering mechanism, we have to include a model of uncertainty $N(\mu_r, \sigma^2_r)$ where μ_r is the reported position of the other robot and σ^2_r is a constant based on the physical size of the robot. This gives rise to the following rule for *a robot position based filtering algorithm* (where $d_r(p)$ is the distance from μ_r to p):

$$Filter(p)\ if:\quad d_r(p) < k\sigma_r \tag{6}$$

Finally, we investigate an alternate model that only involves a robot knowing, with an associated range of uncertainty, the direction to the other robot. We quantify this direction as a line L between the two robot positions and an angle ha^o on each side of L. We introduce the *robot direction based filtering algorithm*:

$$Filter(p)\ if:\ d_p(p, L) \leq d_r(p_L) tan(ha) \tag{7}$$

Where p_L is the (unknown) closest point on L to p, and the distance from p_L to *R1*, $d_r(p_L)$ is given by $d_r(p_L)^2 = d_r(p)^2 - d_p(p,L)^2$.

The performance of each of these four algorithms was measured and is shown in Table 2. Additionally, the best performing map-based approach, frequency filtering, and best performing robot-based approach, direction filtering, were combined and the performance labelled F/D in Table 2. This combined approach reduced mutual view data by 91% and improved the error in fitting the floor plane by 98%.

Table 2: Performance of the four laser-based filtering algorithms and one combined algorithm.

Metric	Freq	Visibility	Position	Direction	F/D Comb
%Ghosts	23	59	93	66	91
%Other	6	19	5	3	20
%Imp. Err	83	48	-17	4	98
%Imp. Out	13	38	4	3	40
%Filtered	12	20	5	4	7.5

Table 3: Timing of the four laser-based filtering algorithms for laser and stereo datasets

Method	Time per point	Laser	Stereo
Frequency	0.23 ms	16.1 s	575 s
Visibility	0.23 ms	16.1 s	575 s
Position	0.32 ms	22.4 s	800 s
Direction	0.27 ms	18.9 s	675 s

3.3 Application to Stereo-camera data

The filtering algorithms developed in section 3.2 were then applied to the data sets generated by the team of two stereo-camera robots. These data sets are bigger than the laser datasets. Where a typical laser dataset was around 70K points, a typical stereo-camera dataset was around 2.5M points. The four approaches were timed running on a Dell Latitude D630 laptop running Ubuntu, and the results shown in Table 3. It is very clear that the methods are too slow to work with stereo

datasets; this was the principal motivation in investigating a new and faster filtering approach based on a model of human visual attention.

4. VISUAL ATTENTION MODEL

The human visual experience is constructed by a mixture of top-down and bottom-up processes [12]. For example, humans are typically unaware of the impact of the 'blind spot' – an area in the retina in which there are no photoreceptors – on their visual experience. Saccadic suppression removes the apparent motion of the visual field caused by fast eye motions scanning the scene. However, the phenomena of most interest from the perspective of our application problem are those that relate to so-called 'looking without seeing' and include change-blindness (the inability to see small changes in a scene), and inattentional-blindness (failure to see objects right in front of the viewer) as well as attentional-blink and repetition blindness. The objective in exploring human visual attention is to build a fast method to filter out mutual view data that works well for stereo-camera datasets as well as the up/down laser datasets.

4.1 Global Workspace Theory

Early work in cognitive psychology (e.g., [12] Chapter 6) explored why humans cannot focus on two streams of information at the same time and proposed a filtering mechanism that blocked the 'ignored' information. Baars [13] proposed a more comprehensive model of selective visual attention, *Global Workspace Theory (GWT)*, illustrated in Figure 4(a). GWT uses the metaphor of a stage: All sensory information is played out onto a mental stage, and all processes that use sensory information do so from the stage. However, only some of that information is consciously attended to, and this is modelled by *spotlights* in GWT: Certain areas of the stage have spotlights directed at them. The spotlights are controlled by a *director*, who can be influenced by top-down concerns as well as local sensory content. Sensory data on the stage outside the fringe of the spotlights is not available to any process that uses sensory data. This is proposed as the mechanism for the 'looking without seeing' phenomena. All the information under the spotlights is available to all processes that use sensory data, and forms a global workspace (hence the name).

To leverage this approach, the problem is recast from filtering data to be ignored (mutual views or 'ghosts') to determining to what to pay attention. For the purpose of this paper, we choose to pay attention to the *walls, ceiling and floor* – since our metric of performance is the fitting of planes to that data. Other choices are possible, depending on the specific application, but this choice is justifiable given that the application is fast mapping of an indoor site by a large heterogeneous team. GWT provides no mechanism for determining how the director decides *where* to place spotlights, only what it does with the spotlight. For example, if the director is to spotlight the walls, floor and ceiling in the sensory data, then it must have a way to identify these and train the spotlight on them. To apply GWT to our problem, a process needs to be added that allows the director to find these regions of interest.

Figure 4(b) shows the modification added to GWT for application to this problem; We will add a preattentive filter with the job of sampling all sensory data and identifying regions for the spotlight to focus visual attention.

4.2 Visual Focus of Attention Filter

The sensory data entering the visual attention filter is the cloud S defined in (1) and which now can include both laser data and stereo-camera data: $S = \{ p \mid p = (r, g, b, x, y, z) \in B^3 \times R^3 \}$, where $B = \{0..255\}$ is the value set of the color components. A visual focus of attention filter, *VFA*, is defined:

$$VFA = (SPF, IBF, \{spf_i\}, \{ibf_j\}) \tag{8}$$

Where SPF and IBF are sets of regions. SPF (Spotlight filter) is a set of spatial regions on which the spotlight will fall and that are likely to contain the walls, floor and ceiling. This is defined in more detail as: $SPF = \{ (x_{mn}, y_{mn}, z_{mn}, x_{mx}, y_{mx}, z_{mx})_I \mid i \in 1..n \}$ specifying the minimum (*mn*) and maximum (*mx*) ordinates for a rectangular region of space. With each region is associated a characteristic function that can be applied to a point in the cloud and returns true iff the point is in the region:

$$spf_i(p) \equiv (x_{mn,I} \le x \le x_{mx,i}) \wedge (y_{mn,i} \le y \le y_{mx,i}) \wedge (z_{mn,i} \le z \le z_{mx,i}) \tag{9}$$

IBF (Inattentional Blindness filter), however, is a set of regions of the color space that will be blocked from visual attention, $IBF = \{ (r_{mn}, g_{mn}, b_{mn}, r_{mx}, g_{mx}, b_{mx})_j \mid j \in 1..m \}$. There is also an associated characteristic function $ibf_j(p)$. Finally, we propose the *Visual Focus of Attention filtering rule*:

$$VFA(S) = S' \text{ where } \forall p \in S,\ p \in S' \text{ iff } (\forall i \in 1..n,\ spf_i(p)) \wedge (\forall j \in 1..m,\ \neg ibf_j(p)) \quad (10)$$

The sets SPF and IBF are constructed by the top-down context and preattentive filter in Figure 4(b). The SPF set was constructed by taking a number ($n=4$) of horizontal and vertical slices (of angle $\alpha=30^o$ and thickness $w=200$ *mm*) of the data spaced equally around the robot. The points in each slice were sorted and the $d=20\%$ most distance points were used to identify the minimum and maximum bounds for a region in SPF. In this paper, IBF just contained one region, and that was constructed by fitting a normal distribution to the color components of a side view of the Pioneer 3-AT and constructing the region to be 1SD on either side of the mean for each color component.

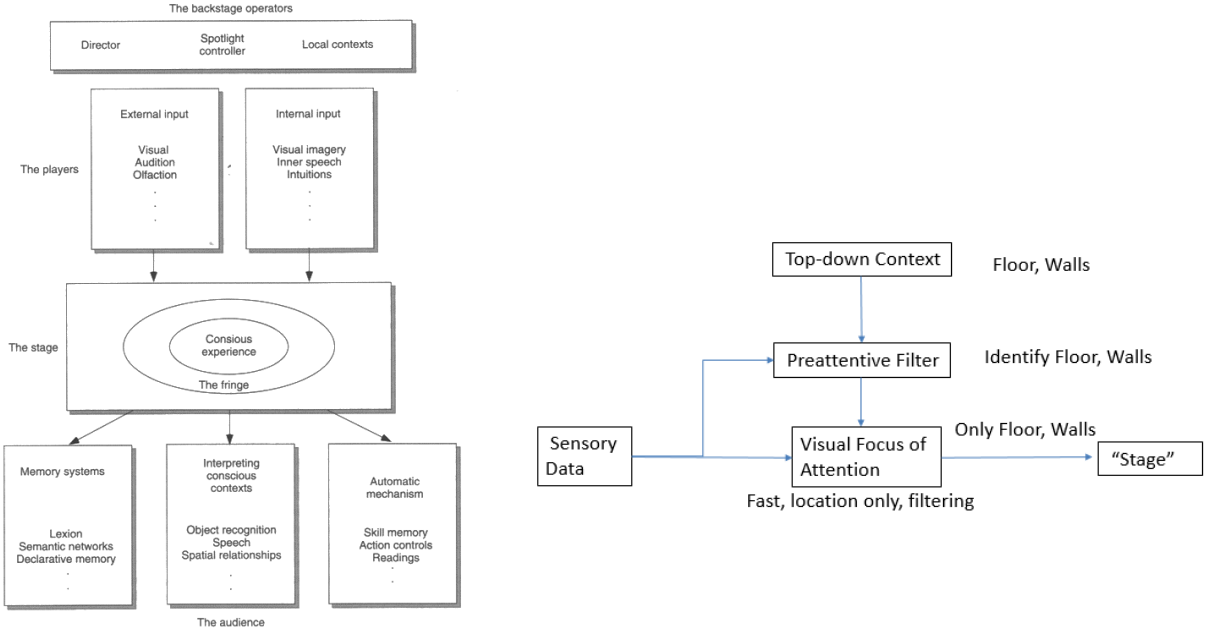

Figure 4: (a) The Global Workspace Theory of selective visual attention (after [12]);
(b) Application to mutual view filtering.

4.3 Results

The VFA filter was applied to two Oval Lap (Figure 5) stereo-camera datasets: VA DS1 has 994 robot locations and produced a final cloud with 1.6M points of which 2.3% were mutual view points to be eliminated; VA DS2 had 1496 robot locations and produced a final cloud of 2.3M points of which 1.15% were mutual view points to be eliminated.

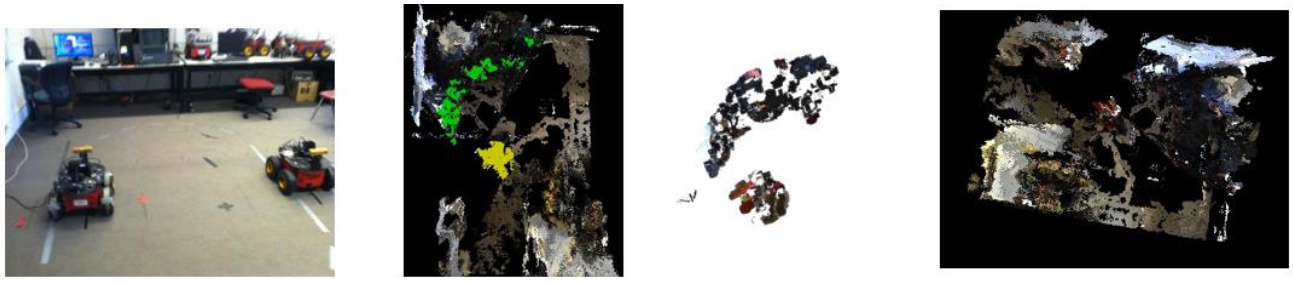

Figure 5: (a) Two Robot Oval Lap configuration; (b) VA DS1 dataset with mutual views pseudocolored; (b) mutual view data from (b) in original color; (d) perspective view of (b) without pseudocolor.

The VFA filtering results are shown in Table 4. This method rejected a large quantity of mutual view data (75% and 70% respectively) and improved the error in fitting the floor plane (32% and 77% respectively). Although good, these numbers

are not as effective as the combined F/D filter in Table 2. However, as expected, the filter operated very quickly (3.66 μs and 3.88 μs respectively) whereas the F/D filter was approximately 100 times slower.

Table 4: VFA Filtering Results

Metric	VA DS1	VA DS2
%Ghosts	75	70
%Other	58	60
%Imp. Err	32	77
%Imp. Out	28	31
Time per point	3.66 μs	3.88 μs
Total time	5.7s	8.7s

5. CONCLUSIONS

When a large team of robots is deployed to a threat site with the objective of constructing a model of the site as fast as possible, within minutes of deployment, and to make the model available for situational awareness at a remote command center, a number of issues beyond those usually considered for SLAM arise. This paper reports our continuing work on one of those topics: the automatic removal of mutual view data from the datasets, what we refer to in the article as 'ghosts', scan data of one robot as seen by a second. For a large team working in a confined area, which is desirable to construct a site model as quickly as possible, this can happen relatively often. However, since it introduces error into the model, this paper studies the issue of rejecting this mutual view data.

In prior work using a two robot team, each configured with a tilted laser scanner so as to generate 3-D scan data with robot motion, we generated a joint dataset that includes mutual view data. The fit of the floor plane to the joint dataset was used as a measure of the effect of the mutual view data, and we showed that an up to 98% improvement in fit was possible if all the mutual view data was manually removed. Four different removal algorithms were evaluated and combination of two removed 91% of the ghost data and produced a 98% improvement in floor plane fit. This was fast for laser datasets, but quite slow for much denser stereo-camera (as well as combined stereo-laser) datasets.

This paper presented a fast filtering approach based on a model of human visual attention. This filter operates hundreds of times faster than the prior filters, achieves 73% removal of mutual view data and 55% improvement in floor plane fit (averaged over two datasets). Our conclusion is that the visual attention approach of deciding what to pay attention to, rather than what to reject, has strong potential for effective mutual view data removal. By focusing on just the walls, floor and ceiling however, much important data is missed. It is important to be able to place that data back into the model once planes have been fitted to the floor, walls and ceiling and this is a topic for future work.

Bibliography

[1] K. Konolige, G. Grisetti, R. Kummerle, B. Limketkai and R. Vincent, "Efficient sparse pose adjustment for 2D mapping," in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2010.

[2] R. Reid and T. Bräunl, "Large-scale Multi-robot Mapping in MAGIC 2010," in *IEEE Robotics, Automation and Mechatronics*, Qingdao China, 2011.

[3] T.-L. Liu and D. Lyons, "Leveraging Area Bounds Information," in *13th Int. Conf. on Intelligent Autonomous Systems*, Padua Italy, 2014.

[4] D. Lyons, K. Shesthra and T.-L. Liu, "Fusion of Ranging Data from Robot Teams Operating in Confined Areas," in *SPIE Defense, Security and Sensing: Multisource, Multisensor Information Fusion*, Baltimore MD, 2013.

[5] F. Lu and E. Milios, "Globally Consistent Range Scan Alignment for Environment Mapping," *Autonomous Robots 4,* 1997.

[6] S. Carpin, "Merging Maps via Hough Transform," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008.

[7] Y. Liu and S. Thrun, "Gaussian Multirobot SLAM," in *Advances in Neural Information Processing Systems, NIPS 2003*, Vancover Canada, 2003.

[8] A. Nuchter, "Parallelization of Scan Matching for Robotic 3D Mapping," in *European Conference on Mobile Robots*, Freiburg Germany, 2007.

[9] R. Kaushik, J. Xiao, W. Morris and Z. Zhu, "3D Laser Scan Registration of Dual-Robot System Using Vision," in *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2009.

[10] C. Bibby and I. Reid, "Simultaneous Localization and Mapping in Dynamic Environments," in *Robotics Science and Systems*, Atlanta GA, 2007.

[11] S. Wangsiripitak and D. Murray, "Avoiding moving outliers in visual SLAM by tracking moving object," in *IEEE Int. Conf on Robotics and Automation*, Kobe Japan, 2009.

[12] J. Friedenberg, Visual Attention and Conciousness, New York: Psychology Press, 2013.

[13] B. Baars, "The global workspace theory of conciousness," in *The Blackwell companion to conciousness*, Malden, Blackwell Publishing, 2007, pp. 236-246.

[14] A. Ortega and J. Andrade-Cetto, "Segmentation of dynamic objects from laser data," in *European Conference on Mobile Robots*, Orebro Sweden, 2011.

[15] M. Piccardi, "Background subtraction techniques: a review," in *IEEE International Conference on Systems, Man and Cybernetics*, The Hague, Netherland, 2004.

[16] D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons and A. Jain, "A Background Model Initialization Algorithm for Video Surveillance," in *IEEE ICCV*, 2001.

[17] K. Litomisky and B. Bhanu, "Removing Moving Objects from Point Cloud Scenes," in *Int. Workshop on Depth Image Analysis*, Tsukuba Japan, 2012.