Multicamera 3D Reconstruction of Dynamic Surgical Cavities: Camera Grouping and Pair Sequencing

Yun-Hsuan Su, Kevin Huang, Blake Hannaford

***Abstract*— Dynamic 3D reconstruction of surgical cavities is essential in a wide range of computer-assisted surgical intervention applications, including but not limited to surgical guidance, pre-operative image registration and vision-based force estimation. According to a survey on vision based 3D reconstruction for abdominal minimally invasive surgery (MIS) [1], real-time 3D reconstruction and tissue deformation recovery remain open challenges to researchers. The main challenges include specular reflections from the wet tissue surface and the highly dynamic nature of abdominal surgical scenes. This work aims to overcome these obstacles by using multiple viewpoint and independently moving RGB cameras to generate an accurate measurement of tissue deformation at the volume of interest (VOI), and proposes a novel efficient camera pairing algorithm. Experimental results validate the proposed camera grouping and pair sequencing, and were evaluated with the Raven-II [2] surgical robot system for tool navigation, the Medtronic Stealth Station s7 surgical navigation system for real-time camera pose monitoring, and the Space Spider white light scanner to derive the ground truth 3D model.**

I. Introduction

Minimally invasive surgery (MIS) allows for smaller incisions, quicker patient recovery, lower risk of infection and less pain [3]. In recent years, medical robots have been incorporated into a variety of surgical procedures to assist surgeons. The surgeons can leverage the high levels of accuracy and dexterity of these robots via teleoperation; instead of manually manipulating surgical tools, surgeons remotely control robot mounted tools through a software layer. While machine robustness is gained, a certain level of human perception and awareness is lost, namely contact and interaction force sensations. This can cause unintentional tissue damage. Because of strict sanitation requirements, electronic tool-mounted sensors are prohibited. An alternative promising approach is vision-based force estimation, whereby interaction force is inferred from visual tissue deformation analysis and an appropriate tissue dynamic model.

For dense 3D reconstruction from RGB cameras, multiple viewpoints are often needed. This can be accomplished via camera motion of a single camera, yet in MIS this is disorienting and distracting. Alternatively, dense surgical scene reconstruction can be pursued from multiple cameras from different viewpoints. Previous work has demonstrated that multiple viewpoint autostereoscopic display (AD) technology maintains stable surgeon perception of the scene while allowing for camera repositioning [4]. This approach allows all cameras to remain relatively motionless while collectively streaming multiple view points. Multiple cameras are particularly amenable to dynamic scenes, not unlike the human body, e.g. caused by respiration and heart beat. This method does not necessitate additional incision ports as cameras can be attached to the interior of the abdomen and provide multiple views once insufflated.

Yun-Hsuan Su and Blake Hannaford are with the University of Washington Department of Electrical and Computer Engineering, 185 Stevens Way, Paul Allen Center - Room AE100R, Campus Box 352500, Seattle, WA 98195-2500, USA. {yhsu83, blake}@uw.edu

Kevin Huang is with Trinity College, Dept. of Engineering, 300 Summit St, Hartford, CT 06106 USA kevin.huang@trincoll.edu

978-1-5386-7825-1/19/$31.00

A. Contribution

The authors' previous publications [5] [6] focus on several components towards vision based force estimation in MIS. This paper extends that body of work to handling multiple camera image viewpoints, as shown in Fig. 1, and proposes novel pairwise sequencing for real-time 3D reconstruction of the dynamic surgical cavity.

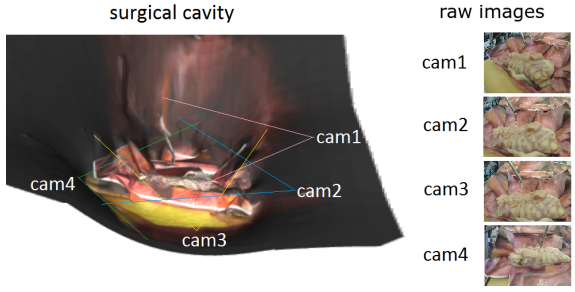

Fig. 1: Illustration of multiple independently moving cameras from different view points looking at the surgical cavity.

Particularly, this paper proposes a graph-based framework for 3D information processing from multiple views in a dynamic environment such that:

- 3D reconstruction is performed efficiently by systematically choosing ideal image pairs.
- Higher robustness is achieved with regard to triangulation error for dynamic surfaces from images captured asynchronously.

B. Related Work

Single moving camera surgical cavity reconstruction approaches have utilized either monocular [7] [8] [9] or stereo [10] [11] [12] vision sensors. With monocular cameras, the extracted 3D shape is represented by a linear combination of predefined basis shapes [13]. Spatial and temporal smoothness constraints were imposed in [14]. [15] [16], followed by relaxing of orthographic assumptions of the camera model.

With stereo vision, [17] extended the factorization approach from [13], and [18] distinguished rigid and moving points based on a global Euclidean transformation check.

Multiple cameras allow for tracking of dynamic tissue shape changes with minimal camera repositioning. Such an approach for MIS was developed for which the cameras were mounted to a single insertable unit through a trocar to avoid multiple additional incision entries [19]. For this, the relative positions of the cameras were fixed. However, recent technical advances in magnetic cameras [20] [21], which can be inserted into the abdominal cavity and controlled by external magnets, can help overcome this limitation. In fact, high precision wireless control of magnetic cameras was achieved for single-incision laparoscopic surgery (SILS) [22] [23]. This technology can allow multiple independently moving cameras that simultaneously record the surgical cavity from multiple viewpoints.

COSLAM achieves visual reconstruction using multiple independent cameras in dynamic environments [24]. However, COSLAM applications are different from MIS settings in several key aspects:

- Camera pose is unknown in COSLAM. However, camera pose can potentially serve as a prior and lead to better reconstruction accuracy.
- COSLAM has been implemented on room-scale environments. Surgical cavities are much smaller.
- Camera motion in COSLAM algorithms do not deviate much from straight-line trajectories. Cameras presented here roughly orbit around the VOI within the surgical cavity.

II. Methods and Experiments

As with in [24], feature matching is conducted across both time and space. While there are many image pairs that can be selected, not all camera pair selections will produce good matching results. This work proposes a carefully designed strategy for image pair selection to increase time efficiency and reduce outliers. Camera matching across time, or intracamera matching, matches feature points among images from the same camera at different time instances. Camera matching across space, intercamera matching, matches feature points from concurrent images from different cameras.

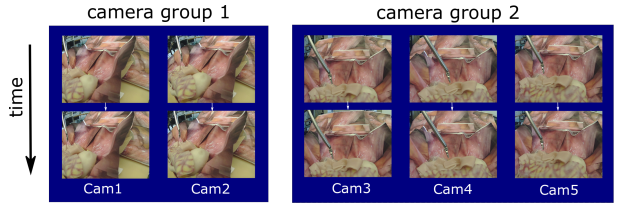

Fig. 2: Camera grouping and (intra/inter)camera matching.

In this method, cameras are grouped together if fields of view (FOVs) overlap by a predefined threshold. Cameras can exist in multiple groups, and features are matched only between cameras within the same camera group. This is conveyed in Fig. 2. This example features five cameras classified into two camera groups. Each cameras undergoes intracamera matching, but only cameras within the same group undergo itercamera matching.

A. Overall Workflow

Since acquired camera images are assumed 2D, 3D reconstruction requires at least two images for comparison and triangulation. These two images can be selected either from the same camera at subsequent time instances or concurrently from two different cameras. In this work, a basic assumption that some 3D points may be generated from an exhaustive search of all possible image pairs.

The static environment case is straightforward. With ideal camera calibrations, computational cost is not a constraint, and with ideal camera triangulation error in reconstruction arises only from feature matching. Finally, assuming a Gaussian model for outlier noises, an average 3D pointcloud of the scene can be generated. However, practical challenges for MIS exist. Time efficiency is a requirement for online applications, so determining the minimum number of image pairs to derive a 3D model is critical. Moreover, in dynamic environments, image pairs from different time instances may be erroneous. The reliability of the generated 3D information needs to be prioritized in this online, dynamic scenario. Fig. 3 illustrates the four sub-tasks within the overall proposed approach, detailed in the following subsections.

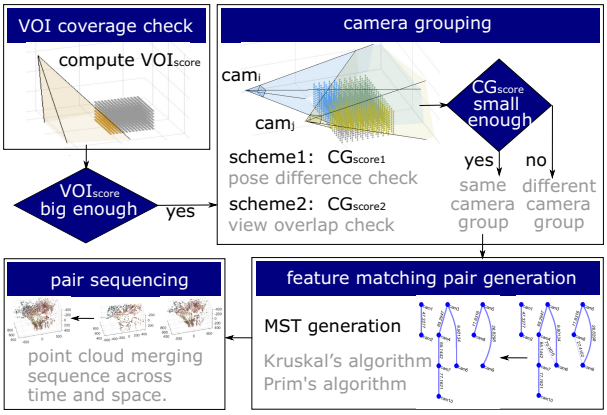

Fig. 3: The workflow for camera grouping and pair sequencing in multiple camera 3D reconstruction of dynamic surgical cavities.

B. VOI Coverage Check

Only cameras with views of the VOI should be considered for grouping and subsequent reconstruction. Some basic assumptions about the cameras allow for a geometric approach to the problem.

Suppose each camera has a rectilinear lens with perspective center at the center of its entrance pupil [25]. A pinhole model then applies, as illustrated in Fig. 4. The camera angle of view (AOV), used interchangeably with FOV [26], describes the angular extent of a given scene that is imaged. The directional components of horizontal, vertical, and diagonal AOV satisfy the following relation:

$$\tan\left(\frac{\alpha_i}{2}\right) = \frac{i}{2 \cdot S_2} \tag{1}$$

where i can take on the designations of h, v, d, the horizontal, vertical or diagonal specification of the image. S_2 is the distance from camera center to image plane.

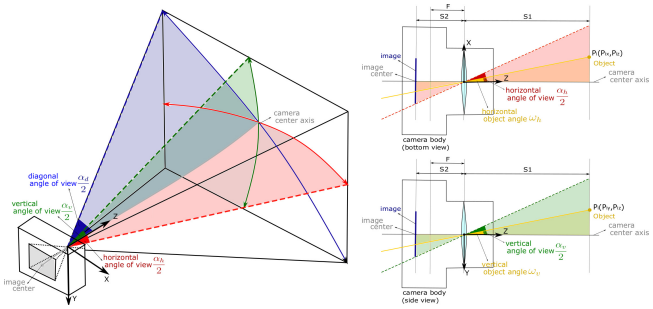

Fig. 4: The horizontal, vertical and diagonal AOVs shown in red, green and blue respectively. S_1 is the object distance from camera, S_2 the distance from camera lens to image plane and F is the camera focus. $S_2 = F$ is required for sharp projection of P_i.

The following are necessary and sufficient conditions for a 3D point, p, to appear within the AOV. The 3D point:

- remains within the horizontal AOV ($\omega_h < \alpha_h$).
- remains within the vertical AOV ($\omega_v < \alpha_v$).
- is the closest point to the camera along the ray cast from camera center to the point.

With these assumptions, a VOI coverage check for each camera ensures that all grouped cameras sufficiently view the VOI. Cameras that fail the check will not be taken into consideration until relocated. First the predefined VOI is discretized into a set of 3D points. The 3D points can be distributed uniformly in space, or in a weighted distribution to emphasize particular regions of the VOI.

Each camera is then scored for VOI coverage, denoted $\text{VOI}_{\text{score}}$. First, let $\vec{i}, \vec{j}, \vec{k}$ represent unit vectors in the camera cartesian frame. Let P be the set of all sampled points of the VOI. Iterating through each point $P_q \in P$, compute the normalized projections onto i, j

$$\sin(\omega_v) = \frac{|\vec{P_q} \cdot \vec{j}|}{|\vec{P_q}|} = \frac{|P_{qy}|}{|\vec{P_q}|} \tag{2}$$

$$\sin(\omega_h) = \frac{|\vec{P_q} \cdot \vec{i}|}{|\vec{P_q}|} = \frac{|P_{qx}|}{|\vec{P_q}|} \tag{3}$$

If the point is within the AOV, coverage is increased. In other words, if $\left(\sin(\omega_h) < \sin\left(\frac{\alpha_h}{2}\right)\right)$ and $\left(\sin(\omega_v) < \sin\left(\frac{\alpha_v}{2}\right)\right)$, then increment

$$\text{VOI}_{\text{score}} = \text{VOI}_{\text{score}} + \frac{1}{q+1}\left(1 - \text{VOI}_{\text{score}}\right)$$

where $P_q = (P_{qx}, P_{qy}, P_{qz})$ is the q^{th} interated point in P, represented in camera frame C with positive depth P_{qz}. Further, ω_h and ω_v are the horizontal and vertical angles between $\vec{k}$ and the ray cast from camera center to P_q. Each camera's $\text{VOI}_{\text{score}}$ represents the VOI coverage and is valued between 0 and 1, 1 representing full coverage. Fig. 5 depicts several $\text{VOI}_{\text{score}}$ values for different configurations, i.e. various geometries and sample point distributions for the VOI. A simple predetermined $\text{VOI}_{\text{score}}$ threshold informs a binary classification distinguishing eligible cameras with enough visibility of VOI from cameras that do not view critical features of interest.

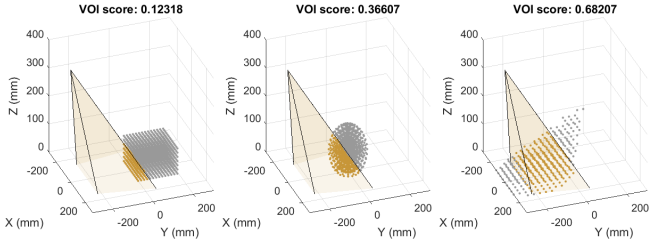

Fig. 5: The $\text{VOI}_{\text{score}}$ values for three different VOI sample point distributions. Sample points from left to right: (1) uniformly distributed in a cube (2) uniformly distributed in the spherical coordinate system (azimuth, elevation, radii), points appear denser near the center in Cartesian space; (3) distributed along a cone shape similar to the tool range of motion under the Remote Center of Motion (RCM) constraint.

As an example, consider a set of ten cameras within a surgical cavity. These cameras provide various viewpoints within the cavity, and their camera poses are known apriori. This scenario is depicted below in Fig. 6.

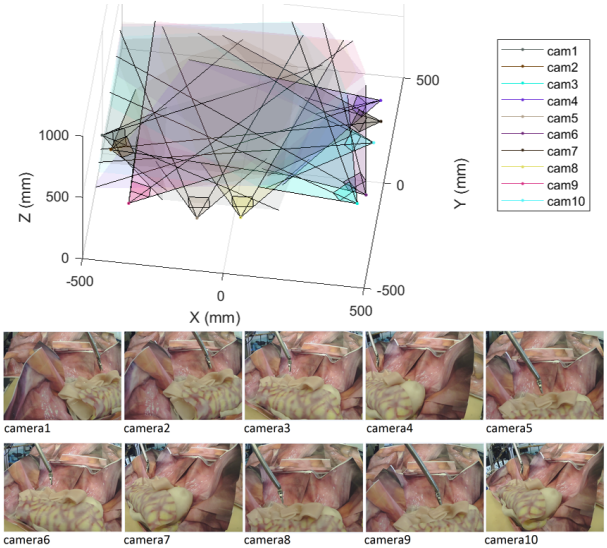

Fig. 6: This is a set of 10 images from different camera viewpoints and the visualization of the camera poses.

The $\text{VOI}_{\text{score}}$ can then be calculated for each camera given a known workspace geometry and VOI sampling distribution. For this, a cubic workspace and uniform sampling is chosen, and $\text{VOI}_{\text{score}}$ values are determined as shown in Fig. 7.

C. Camera Grouping

Of the cameras with sufficient $\text{VOI}_{\text{score}}$, camera groupings for optimal feature matching must be formed. This is achieved through a graph-based approach and another metric, CG_{score}. A fully connected graph is constructed with each vertex a camera and each edge weighted by a CG_{score}. Edges with weights greater than a CG_{score} threshold are broken. Subsequently, initial non-overlapping camera groups are formed as the remaining isolated sub-graphs. These are divided into mutually overlapping camera groups by dividing

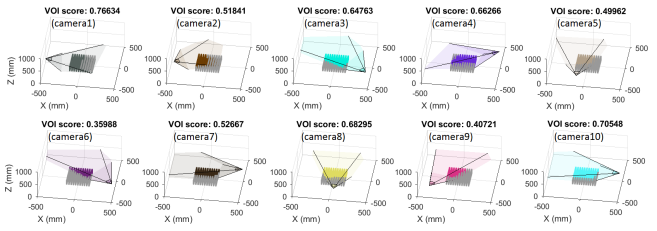

Fig. 7: $\text{VOI}_{\text{score}}$ values for all 10 cameras from Fig. 6. VOI defined as a cubic workspace spanning $X, Y = [-150, 150]$ and $Z = [0, 150]$ and with uniform sampling.

into the largest complete sub-graphs. A critical component of this procedure is the calculation of CG_{score}. Two proposed methods for described, based on (1) pose difference and (2) view overlap.

1) CG_{score1} – Pose Difference: This score simply compares the relative configuration between two cameras. Let $\vec{t}_{ij}$ denote the translation between camera i coordinate frame and camera j. Also, suppose R_{ij} is the relative rotation between cameras i and j. Then define

$$\text{CG}_{\text{score1}}(i,j) = \left|\left|\vec{t}_{ij}\right|\right|_2 + \left|\cos^{-1}\left(\frac{\text{tr}(R_{ij}) - 1}{2}\right)\right|$$

which ranges from 0 to ∞; smaller values of $\text{CG}_{\text{score1}}$ indicate more similar camera viewpoints. Camera groups using $\text{CG}_{\text{score1}}$ for the scenario depicted in Fig. 6 are shown in Fig. 8

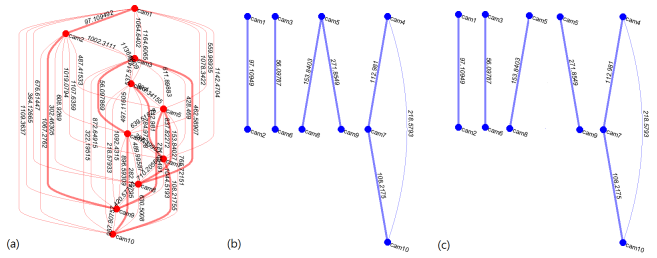

Fig. 8: (a) $\text{CG}_{\text{score1}}$ of each camera pair (b) and (c) are the non-overlapping and overlapping camera grouping result.

2) CG_{score2} – View Overlap: This grouping scheme inherits the previously computed $\text{VOI}_{\text{score}}$ value utilizes the Sørensen-Dice index [27]. Again, let P denote the set of sampled VOI points. For arbitrary camera i, let Cam_i be the the subset of P viewable by camera i, as determined by (2) and (3). The quantifiable metric $\text{CG}_{\text{score2}}$ is then defined as:

$$\text{CG}_{\text{score2}}(i,j) = -\frac{2\left|\text{Cam}_i \cap \text{Cam}_j\right|}{\left|\text{Cam}_i\right| + \left|\text{Cam}_j\right|} + 1$$

where $|.|$ denotes cardinality. $\text{CG}_{\text{score2}}$ ranges from 0 to 1; smaller values of $\text{CG}_{\text{score2}}$ indicate more similar camera viewpoints. Camera groups using $\text{CG}_{\text{score2}}$ for the scenario depicted in Fig. 6 are derived and subsequently depicted in Fig. 9

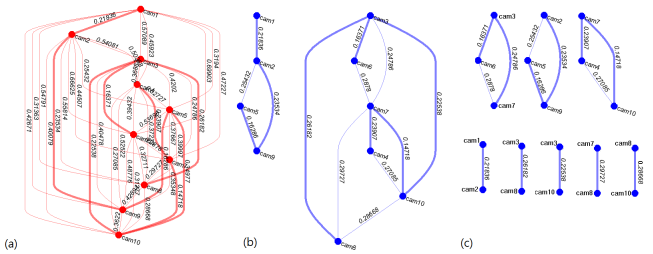

Fig. 9: (a) $\text{CG}_{\text{score2}}$ of each camera pair (b) and (c) are the non-overlapping and overlapping camera grouping result.

3) Threshold Derivation: The camera grouping threshold is determined by Algorithm 1 for graph G [28]:

Algorithm 1 $\text{Thresh}_{\text{CG}}(\text{G})$

set s as threhold increment step size
set $r_{\text{best}} = \infty$
set $e_{\min}$ = least edge weight in G
set th_c = greatest edge weight in G
set $th_{\text{best}} = th_c$
while $th_c > e_{\min}$ **do**
    remove edges with weight $\geq th_c$
    calculate weight ratio for remaining graph, r
    if $(r < r_{\text{best}})$ **then**
        $r_{\text{best}} = r$
        $th_{\text{best}} = th_c$
    end if
    $th_c = th_{\text{best}} - s$
end while
return th_{best}

where r calculated in step 8 is the ratio between the mean intra-cluster edge weight and mean inter-cluster edge weight.

In Fig. 8-(b), four camera groups are generated with a threshold of 280, separating camera pairs with large $\text{CG}_{\text{score1}}$. In this case, each camera belongs to exactly one camera group. Fig. 8-(c) illustrates overlapping camera groups where a camera can exist in multiple camera groups simultaneously, which is achieved by mandating a complete graph. A threshold of 0.3 for $\text{CG}_{\text{score2}}$ results in the camera groups depicted in Fig. 9. The viewpoint overlaps for the scenario depicted in Fig. 6 using the two proposed CG_{score} algorithms can be visualized graphically, as shown in Fig. 10.

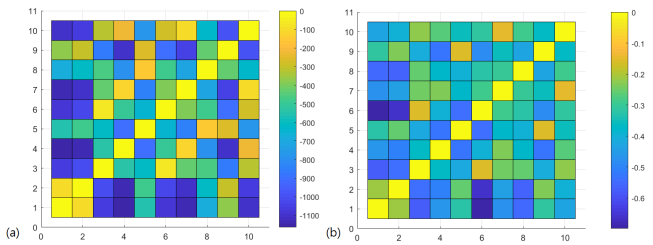

Fig. 10: View overlap using procedures detailed in II-C.1 and II-C.2. On the left is $-\text{CG}_{\text{score1}}(i,j)$ and on the right is the $-\text{CG}_{\text{score2}}(i,j)$ scores. The x and y axes are camera indices. Warmer colors indicate more view overlap.

D. Feature Matching Pair Generation

After camera groups are formed, optimal camera pairs for triangulation with groups must be determined; using all pairs can be redundant. To facilitate this, a minimum spanning tree (MST) approach is utilized. Two popular methods exist to find the MST: Kruskal's algorithm and Prim's algorithm [29]. The MST results are distinguished by thicker edges in Fig. II-C.1 and Fig. II-C.2. Table I shows the average run time for both algorithms using the various graphs and CG_{score} values.

Each MST algorithm will be briefly explained given $N \in \mathbb{N}$ cameras, and two undirected graphs $G_1(V, E_1), G_2(V, E_2)$ where vertices V is the set of N cameras and the edges E_1, E_2 are defined by the two camera group scores, $\text{CG}_{\text{score1}}$ and $\text{CG}_{\text{score2}}$. Specifically, $E_1 = \{\text{CG}_{\text{score1}}(i,j)\}$ and $E_2 = \{\text{CG}_{\text{score2}}(i,j)\}$ for $i, j \in \{1, 2, ..., N\}$. Section II-C detailed decomposing G_1 and G_2 into sub-graphs via heuristically tuned CG_{score} thresholds. The following subsections describe MST derivation for sub-graph $g(v, e)$ using Kruskal's and Prim's algorithms.

1) MST via Kruskal's: In Kruskal's algorithm, the MST structure begins by selecting the edge with minimum weight, as determined by CG_{score}. The remaining edges are added one-by-one to the MST structure based on edge weight, so long as their addition does not create a closed loop in the MST. Once all nodes are included, the MST is complete.

2) MST via Prim's: In Prim's algorithm, the MST structure begins by arbitrarily selecting a vertex of the sub-graph, and is removed from the set of remaining vertices. From the remaining vertices, the vertex connected to any element in the MST with least edge weight, as determined by CG_{score} is added to the MST and removed from the pool of remaining vertices. This process is continued until all vertices are added to the MST, at which point the MST is complete.

Graph			Runtime [ms]	
Figure	Overlap	CG_{score}	Kruskal's	Prim's
Fig. 8-(a)	-	$\text{CG}_{\text{score1}}$	4.7123	4.0939
Fig. 8-(b)	no	$\text{CG}_{\text{score1}}$	5.0104	5.4402
Fig. 8-(c)	yes	$\text{CG}_{\text{score1}}$	5.3571	5.9124
Fig. 9-(a)	-	$\text{CG}_{\text{score2}}$	3.9528	3.2198
Fig. 9-(b)	no	$\text{CG}_{\text{score2}}$	3.8232	3.4516
Fig. 9-(c)	yes	$\text{CG}_{\text{score2}}$	4.0016	4.6392

TABLE I: Mean run time for camera feature matching pair generation using different MST algorithms and weighting functions over ten trials. Blue denotes the better performing MST algorithm for the given test condition. Note: ZNCC and ORB are adopted respectively for feature matching and feature points extraction.

E. Pair Sequencing in Time and Space

The 3D model of the surgical cavity is obtained by merging point clouds derived from triangulating matched feature points from every generated image pairs, including both intra- and intercamera pairs. The intercamera pairs are pairs of connected vertices in the MST. However, the accuracy of the final 3D model is affected by the merging sequence. Severla considerations are made in determining both intra- and intercamera pair sequencing. The methods used in this work are described in the following subsections.

1) Intercamera Pair Matching:

a) Sub-Graph Point Cloud Generation: At this point each camera group is represented by the MST of some sub-graph g. Select arbitrarily s, a leaf vertex in g. An initial point cloud is generated from s and an adjacent vertex. Subsequent point clouds are merged by traversing the remainder of the MST via a depth first search approach. Each merge step contributes to a cumulative point cloud, PC_{gi} where i denotes the merge iteration. A final point cloud is generated for sub-graph g, denoted PC_g. To ensure tolerance of map point position uncertainties, a Gaussian feature detection error $N(0, \sigma^2 I)$ is imposed for every merged point.

b) Reprojection Error Check: Prior to each merge step i in sub-graph point cloud generation, PC_{gi-1} is reprojected to each image frame in g. Only the points whose maximum reprojection error REP_{err}, which is simply the Mahalanobis distance between reprojected point and nearest feature point within each camera frame [30].

2) Intracamera Pair Matching: Intracamera matching classifies dynamic vs static areas of PC_g. For each camera vertex within sub-graph g, REP_{err} of points in PC_g is calculated and compared between the current and previous camera frame (in this work, a framerate of 60 Hz is utilized). If the difference in reprojection error is greater than a predefined threshold, the point is labeled as non-static.

3) Fused Camera Group Result: The generated point clouds from each camera group are combined to form an aggregate surgical cavity 3D model. Because camera groups were formed based on workspace visibility, limited overlap will occur between camera group point clouds. Where overlap does occur, uncertainty correction as described for sub-graph point cloud generation is employed.

F. Experimental Setup

Data were collected from ten tracked, arbitrarily moving cameras viewing a phantom surgical scene for three minutes. The frame rate for all cameras was 60 Hz. A total of six test cases of interest were generated to evaluate different grouping parameters. First, selection of camera grouping method could be classified into four grouping methods: exhaustive pair grouping, nearest pair grouping, $\text{CG}_{\text{score1}}$, or $\text{CG}_{\text{score2}}$. The latter two methods are divided further into group overlap or no group overlap, as determined via CG_{score} threshold described in Section II-C – if cameras are allowed to exist in multiple camera groups, overlap is present.

Each experimental condition was evaluated against three metrics of interest, which reflect the density of reconstruction, reconstruction error, and algorithm efficiency respectively:

1) total number of points generated in the 3D surface reconstruction
2) RMSE from ground truth
3) number of camera pairs evaluated

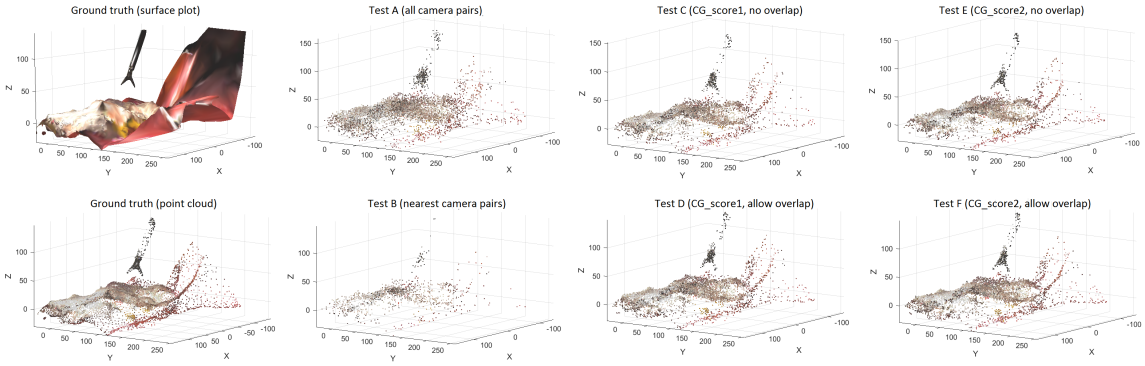

Fig. 11: 3D reconstruction results using different camera groupings schemes.

III. Results

Fig. 11 depicts the final surgical cavity 3D reconstruction results from each test condition, and the evaluated metrics are shown in Table II.

Test	Pairing	Overlap	N	RMSE [mm]	Pairs
A	every pair	-	8988	4.2773	45
B	nearest pair	-	1382	2.4424	9
C	CG_{score1}	N	6457	1.7359	6
D		Y	7349	1.6867	6
E	CG_{score2}	N	7056	1.5529	8
F		Y	8678	1.2887	11

TABLE II: Experimental results showing for each test condition: N - number of points generated, RMSE - error from ground truth point cloud, Pairs - number of camera pairs evaluated. The time efficiency is roughly proportional to the number of triangulated camera pairs. Note: there are N= 16870 points in the ground truth point cloud.

Consider the following observations:

- **Test A**: Exhaustive camera pairing results in a dense, noisy point cloud. Since no grouping was performed, reprojection error conditioning as described in Section II-E.1.b is unfeasible, resulting in large N.
- **Test B**: Cameras are paired with nearest neighbor, resulting in one single camera group. The resultant point cloud is of higher precision but sparser.
- **Test C-F**: The proposed camera grouping methods resulted in denser, less noisy and more efficient point clouds than Test A. Tests D and F allow camera group overlap which leads to higher point cloud density and accuracy, yet may cost computational time.
- **Test C,D**: CG_{score1} is faster to compute and is robust to workspace size. In contrast, CG_{score2} exhibits runtime proportional to workspace size and sample density (as shown in Fig, 5), and is less suitable in larger of time-varying/dynamic workspaces. In this particular experiment, the number of camera pairs does not change between Test C and Test D. Difference in camera groupings result in variation in N and $RMSE$.
- **Test E,F**: View overlap results in more matched feature points, and thus CG_{score2} fits the problem objective well. In this condition, camera group overlap has a greater effect on all three metrics as compared with CG_{score1}.

IV. Conclusion

Dynamic 3D reconstruction of surgical cavities is crucial for MIS. Multicamera approaches are promising. Toward that end, this work presented two novel graph-based camera grouping schemes - overlapping and non-overlapping, where edge weights (and thus groupings) are determined by CG_{score1} or CG_{score2}. Camera pairings are determined via an MST approach, while fusion and classification of dynamic areas was achieved via pair sequencing accross time and space. The results from the experiment show improvements in point cloud accuracy and computational efficiency.

Future work aims to extend and enhance principles developed here towards real surgical scenarios, as opposed to experimental phantoms. For one extension, in the application of a MIS, the VOI can be viewed as a cone extending from the RCM of the surgical robot. By relaxing the camera FOV to a cone geometry, the view overlap between two cameras in the VOI reduces to a volume that is the intersection of three overlapping cones. This geometric problem is addressed in [31], and could imply that CG_{score2} computation can be invariant of VOI size. This is a potential future direction.

Furthermore, since surgical cavities are not always concave, occlusions can prove to be an issue. In particular, good feature points may not be guaranteed despite promising CG_{score2} values; it is not robust to severe occlusion. Thus, the curvature of the environment should be mathematically incorporated in the CG_{score} evaluation function once an initial 3D topology of the scene is generated from the early stages of the image sequence.

Finally, vision-based force estimation in MIS requires an analysis of tissue deformation. Thus, while the classification of dynamic and static regions in the surgical cavity is a step in the right direction, dynamic points should be further identified as deforming or merely translating surfaces. One preliminary approach could involve thresholding the motion derivative of a dynamic point in its neighborhood. Stark variation of motion derivatives across a continuous surface region indicate deformation over translation.

Acknowledgements

The authors would like to thank Fangbo Qin for assisting with the data collection and the WISH Lab, Harborview Medical Center, Seattle for offering the experiment platform.

REFERENCES

[1] B. Lin, Y. Sun, X. Qian, D. Goldgof, R. Gitlin, and Y. You, "Video-based 3d reconstruction, laparoscope localization and deformation recovery for abdominal minimally invasive surgery: a survey," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 12, no. 2, pp. 158–178, 2016.

[2] B. Hannaford, J. Rosen, D. W. Friedman, H. King, P. Roan, L. Cheng, D. Glozman, J. Ma, S. N. Kosari, and L. White, "Raven-II: an open platform for surgical robotics research," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 4, pp. 954–959, 2013.

[3] K. Fuchs, "Minimally invasive surgery," *Endoscopy*, vol. 34, no. 02, pp. 154–159, 2002.

[4] H. Urey, K. V. Chellappan, E. Erden, and P. Surman, "State of the art in stereoscopic and autostereoscopic displays," *Proceedings of the IEEE*, vol. 99, no. 4, pp. 540–555, 2011.

[5] Y.-H. Su, K. Huang, and B. Hannaford, "Real-time vision-based surgical tool segmentation with robot kinematics prior," in *Medical Robotics (ISMR), 2018 International Symposium on*. IEEE, 2018, pp. 1–6.

[6] Y.-H. S. Isaac Huang, Kevin Huang and B. Hannaford, "Comparison of 3d surgical tool segmentation procedures with robot kinematics prior," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2018*, October 2018.

[7] O. G. Grasa, J. Civera, and J. Montiel, "Ekf monocular slam with relocalization for laparoscopic sequences," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4816–4821.

[8] T. Collins, B. Compte, and A. Bartoli, "Deformable shape-from-motion in laparoscopy using a rigid sliding window." in *MIUA*, 2011, pp. 173–178.

[9] M. Hu, G. P. Penney, D. Rueckert, P. J. Edwards, F. Bello, R. Casula, M. Figl, and D. J. Hawkes, "Non-rigid reconstruction of the beating heart surface for minimally invasive cardiac surgery," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2009, pp. 34–42.

[10] B. Lin, A. Johnson, X. Qian, J. Sanchez, and Y. Sun, "Simultaneous tracking, 3d reconstruction and deforming point detection for stereoscope guided surgery," in *Augmented Reality Environments for Medical Imaging and Computer-Assisted Interventions*. Springer, 2013, pp. 35–44.

[11] P. Mountney and G.-Z. Yang, "Motion compensated slam for image guided surgery," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2010, pp. 496–504.

[12] M. Lourenço, D. Stoyanov, and J. P. Barreto, "Visual odometry in stereo endoscopy by using pearl to handle partial scene deformation," in *Workshop on Augmented Environments for Computer-Assisted Interventions*. Springer, 2014, pp. 33–40.

[13] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering non-rigid 3d shape from image streams," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2. IEEE, 2000, pp. 690–696.

[14] M. Paladini, A. Bartoli, and L. Agapito, "Sequential non-rigid

[19] M. Silvestri, T. Ranzani, A. Argiolas, M. Vatteroni, and A. Menciassi, "A multi-point of view 3d camera system for minimally invasive

structure-from-motion with the 3d-implicit low-rank shape model," in *European conference on computer vision*. Springer, 2010, pp. 15–28.

[15] J. Xiao and T. Kanade, "Uncalibrated perspective reconstruction of deformable structures," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 2. IEEE, 2005, pp. 1075–1082.

[16] R. Hartley and R. Vidal, "Perspective nonrigid shape and motion recovery," in *European Conference on Computer Vision*. Springer, 2008, pp. 276–289.

[17] A. Del Bue and L. Agapito, "Non-rigid 3d shape recovery using stereo factorization," in *Asian Conference of Computer Vision*, vol. 1, 2004, pp. 25–30.

[18] X. Lladó, A. Del Bue, A. Oliver, J. Salvi, and L. Agapito, "Reconstruction of non-rigid 3d shapes from stereo-motion," *Pattern Recognition Letters*, vol. 32, no. 7, pp. 1020–1028, 2011.
surgery," *Sensors and Actuators A: Physical*, vol. 202, pp. 204–210, 2013.

[20] G. Yin, W. K. Han, S. Faddegon, Y. K. Tan, Z.-W. Liu, E. O. Olweny, D. J. Scott, and J. A. Cadeddu, "Laparoendoscopic single site (less) in vivo suturing using a magnetic anchoring and guidance system (mags) camera in a porcine model: impact on ergonomics and workload," *Urology*, vol. 81, no. 1, pp. 80–84, 2013.

[21] M. Morgan, E. O. Olweny, and J. A. Cadeddu, "Less and notes instrumentation: future," *Current opinion in urology*, vol. 24, no. 1, pp. 58–65, 2014.

[22] P. Valdastri, C. Quaglia, E. Buselli, A. Arezzo, N. Di Lorenzo, M. Morino, A. Menciassi, P. Dario *et al.*, "A magnetic internal mechanism for precise orientation of the camera in wireless endoluminal applications," *Endoscopy*, vol. 42, no. 6, p. 481, 2010.

[23] N. Di Lorenzo, L. Cenci, M. Simi, C. Arcudi, V. Tognoni, A. L. Gaspari, and P. Valdastri, "A magnetic levitation robotic camera for minimally invasive surgery: Useful for notes?" *Surgical endoscopy*, vol. 31, no. 6, pp. 2529–2533, 2017.

[24] D. Zou and P. Tan, "Coslam: Collaborative visual slam in dynamic environments," *IEEE transactions on pattern analysis and machine intelligence*, 2012.

[25] D. A. Kerr, "The proper pivot point for panoramic photography," 2008.

[26] T. Dobbert, *Matchmoving: the invisible art of camera tracking*. John Wiley & Sons, 2006.

[27] I. Laina, N. Rieke, C. Rupprecht, J. P. Vizcaíno, A. Eslami, F. Tombari, and N. Navab, "Concurrent segmentation and localization for tracking of surgical instruments," *CoRR*, vol. abs/1703.10701, 2017.

[28] P. K. Jana and A. Naik, "An efficient minimum spanning tree based clustering algorithm," in *Methods and Models in Computer Science, 2009. ICM2CS 2009. Proceeding of International Conference on*. IEEE, 2009, pp. 1–5.

[29] A. Kershenbaum and R. Van Slyke, "Computing minimum spanning trees efficiently," in *Proceedings of the ACM annual conference-Volume 1*. ACM, 1972, pp. 518–527.

[30] G. J. McLachlan, "Mahalanobis distance," *Resonance*, vol. 4, no. 6, pp. 20–26, 1999.

[31] F. A. Balogun, A. Brunetti, and R. Cesareo, "Volume of intersection of two cones," *Radiation Physics and Chemistry*, vol. 59, no. 1, pp. 23–30, 2000.