VR Surgery Registration Pipelines ICRA Proceedings

Ruixing Liang
Johns Hopkins University
Baltimore
rliang7@jh.edu

Hongchao Shu
Johns Hopkins University
Baltimore
hshu4@jhu.edu

Abstract

Surgical simulators have been recently exploited not only as a platform to plan and train surgery preoperatively but also as a tool for data generation. However, attempts to develop algorithm in virtual simulation when placed in real scenario tended to perform worse than expected. While efforts has been made to either get more accurate real world data or more natural simulated data, using partly from both has not been investigated. Real Data alone involves cascaded calculations and transformations which intuitively amplify the error throughout each step. Therefore, we proposed a novel compositional paradigm along with AMBF++ of real world data and generation of virtual reality annotations to refine and expedite the data generation process both in favor of better surgical training and preoperative plan but more importantly for overcoming the data scarcity issue existed wide-spread in Deep Neural Networks (DNNs) training. In particular, our framework is demonstrated to achieve down to 0.6mm in surgical tool translation error furthermore with down to 1 mm and 1 ° error in pose tracking to achieve the state-of-the-art performance in this specific task. Our code and sample data are released at https://github.com/Rxliang/RoboMaster

1. Introduction

The dynamic nature of surgical procedures demands sophisticated judgement, professional experiences and high level of attentions. Training and Evaluation of surgeon initially requires apprenticeships which sacrifices time, resources, patient's satisfaction and subsequently contributes to possibility of complications. [2]. This gave rise to the Virtual Reality (VR) simulators, which is considered to be a cost-effective way to provide surgeons with efficiency flexibility and customizability. [6]. Recently, with the burgeon of DNNs and other novel algorithm solutions, the data scarcity of surgical data becomes more prominent and the evaluation platform also seems to be not enough. VR simulators, in this way, has been begun to be recognized as one of tantalizing means to generate masked data and perfect platform field for novel algorithms [8]. However, the effectiveness has been questioned when it comes to adaptation from simulator environment to real world scenario since the naturalness of generated data. Therefore, efforts has been made in order to improve the data naturalness in other computer vision realm, like self-driving car in outdoor uncontrolled complex environment [7] or navigation in relatively controlled in door environment [9]. Alternatively, attempts to narrow the gap between the virtual simulator and real world by applying domain adaptation has been also investigated. [1].

Though these attempts has successfully demonstrate their availability and performances through certain benchmarks in different tasks, limitations still existed if the evaluation on naturalness paradigm to distinguish virtual and real world could not be properly proposed. As we consider, the ill posed problem is the result of using both data and label for convenience, in particular, VR generated data with the corresponding multimodal labels . So instead of utilizing them in one time, we proposed to use real images with close to perfect label generated from VR simulator (around threshold of human perception accuracy) to circumvent the issue of unnatural images or domain mismatch problem mentioned above.

By unifying not only surgical training input and data generation features existed in recent work [8], but also real world data stream from well-calibrated and registered using state-of-the-art algorithms, we propose a upgrade version of AMBF+ close to real VR simulator AMBF++ for both human training and more importantly algorithm development and data generation platform. This framework responds to previous work's limitation and overcomes the aforementioned drawbacks of current practices of data generation in surgical VR realm specifically and set up a new paradigm.

To evaluate our the effectiveness of AMBF++ in this regard, we developed series of unit test on each component, and demonstrated its functionality in specific surgical tasks in skull-base surgery called mastoid surgery. During this

surgery, surgeons are expected to inspect the region of interest(ROI) and use surgical tools in this case drill to remove bone to expose ROI more clearly and manage to get access to certain anatomy structure [11]. Real time and simultaneous tracking of both surgeon ROI and surgical tool is of great significance to computer assisted surgery. However, existing tracking solution is disruptive in current surgical workflow especially within the scope of our focus mastoid surgery. An urge to tailor a stereo video-based tracking for accurate surgical navigation system has been developing recently harnessing the power of DNNs. The data scarcity in this specific surgery could not be properly addressed in previous research which serves as a great downstream sample tasks for AMBF++. We further elaborate the analysis of results and discuss the future directions at last.
The prominent aspects of our work could be summarized as following keynotes:

- A new compositional data generation paradigm to utilize real world surgical image data and generated corresponding label from VR simulator for DNNs training to solve data scarcity problem.
- A calibration and registration pipelines, as fundamental process for real world data acquisition, is provided with benchmark to standardize current rather fragmentation of works to enable later researchers and users to quick start optical tracker working seamlessly with microscope.
- A VR simulator upgraded from AMBF++ with docker file is designed to give cross-platforms consistent user experiences with enabled Xserver GUI implementation. This will enable users to have access to real world data input stream besides original haptic and VR headset input.

2. Methods and Unit Evaluation

2.1. Overall Set Up

Our project was built upon AMBF+ [8], which has already successfully combined multiple state-of-the-art open sourced libraries to realize most of modules we need except for real-world data acquisition. To achieve that, we require more peripherals than original AMBF+.1

Item Name	Model Details	Usage
Optical Tracker	Atracsys Fusion track 500	Collect 3D Coordinates
Optical marker * 13	Atracsys Inc. NDI Inc.	Work with above together
Microscope Camera	HS Allegra 500 & Lenovo Desktop	Collect video flow of surgery procedure
Phantom	Phacon Inc.	Anatomy ROI to be tracked
Surgical Driller	J& J Inc.	Tool to be tracked
Pointer and Frames	NDI Inc. and Housemade	Rigid connect marker with objects to track
Laser Printed Calibration Target with Rigid Substrate	Charuco Target & Checkerboard Target	Calibrate Camera
VR Headset	Valve Inc.	Visualize VR Environment

Figure 1. Required Hardware

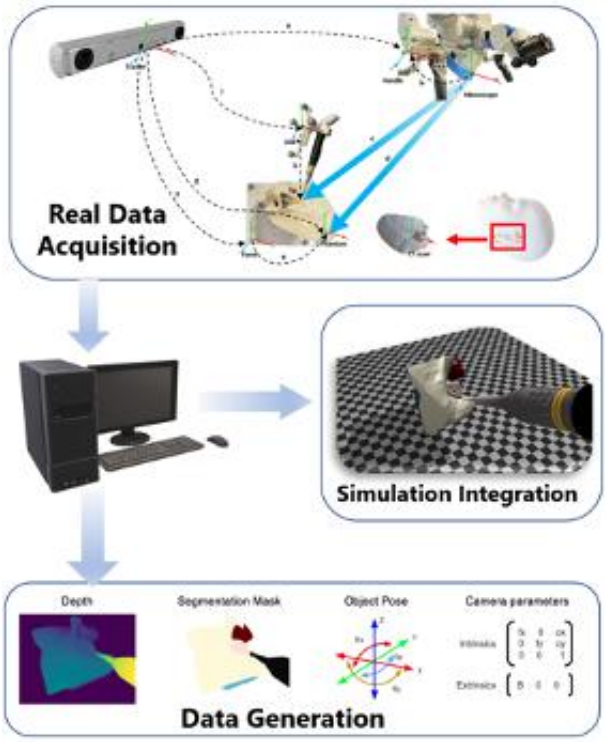

Figure 2. Overall Setup

Additional pipelines has been implemented to get 3D poses of objects of interests in real time, in our case, surgical tools driller tip, phantom (Phacon) and microscope handler using optical tracker and video stream of ROI using microscope. After refined registration of different coordinates, pose of driller tip, phacon in camera coordinates will be transmitted to computer and it will produce the corresponding VR scene meanwhile generate data of significance to pose estimation training.2 In later sections, methods we implemented and benchmarked in each step of the registration process will be addressed in detail.

2.2. Optical Tracker

To setup Atracsys fusionTrack 500, Ubuntu with ROS1 is recommended. Download and compile the Atracsys SDK 4.5.2 from atracsys website, and properly compile cisst/sawAtracsysFusionTrack. Then define the geometry and configuration files for optical frames. For us, we defined geometry for the surgical drill, phantom platform, and microscope.
Though documentation of calibrated optical tracking system of Atracsys is less than 0.1 mm within our distance setting, as the unit evaluation, we still designed an experiment to verify the accuracy of Atracsys within its working volume. We fixed the Atracsys, change the relative orientation of a micrometer stage for 3 times as shown 3.

We moved 5 mm along x, y, z axis respectively with each micrometer stage orientation, then count the frequencies of travel distance for each orientation 4. The worst case has $distance = 5.033 \pm 0.1mm$

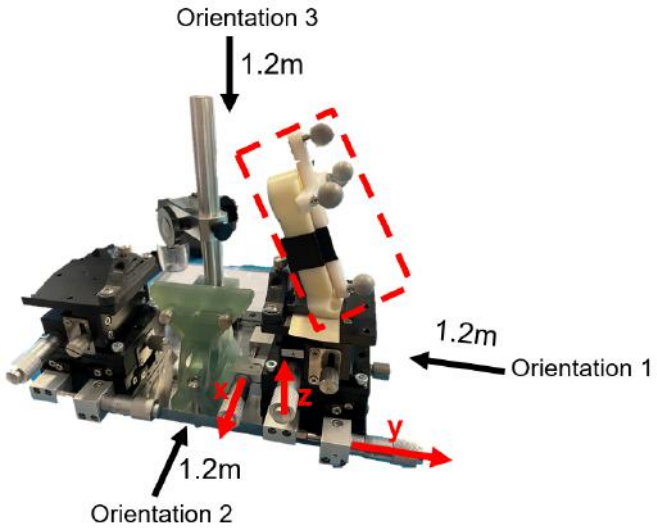

Figure 3. Atracsys Evaluation Setup

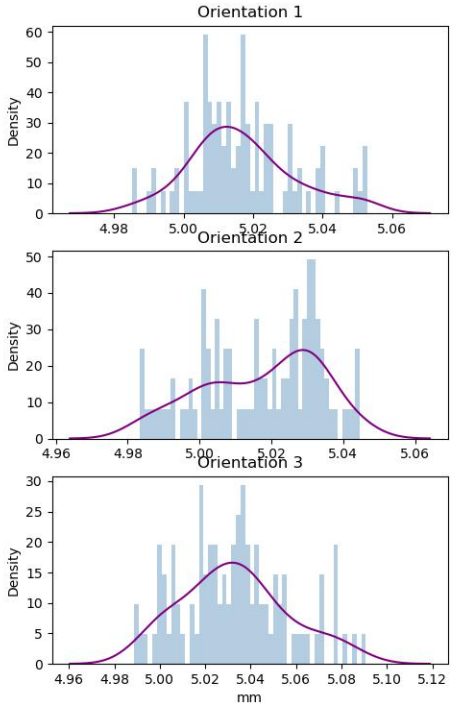

Figure 4. Atracsys Evaluation Result

2.3. Camera Calibration Stereo Rectification and Video Deinterlace

Camera Calibration is of great importance to stereo-microscopic vision system. [10] Each optical microscope should go through delicate calibration process before user use them for computer vision tasks, with certain evaluation process typically speaking, including reprojection error (RPE) and Epipolar error. In general, the existing calibration methods could be classified into two categories [3], including self calibration requires accurate knowledge of motion of cameras, whereas conventional methods used fixed camera, a looser condition for calibration. The crucial component in conventional camera calibration approach is the patterns selections, 3D and 2D patterns are available for calibrations. 3D patterns only requires one shot, a direct linear transformation algorithm is all we need to calibrate the camera. The apparent limitation is that it is expensive to fabricate an accurate 3D micro-scale cube. As Chen et al. Decided, planar pattern based calibration should be the most cost effective way to calibrate cameras. A homographic transformation matrix maps 2D mark points of the pattern to 2D image points at each position. Based on the invariance of the camera's intrinsics, a factorization method can be devised to decompose the constraints provided by several homographic matrices. Two main stream target have been investigated in our experiments.

From our observation, the printing quality will influence the calibration results as well as the substrate rigidity and flatness of the target. Though we could not purchase a well fabricated target in time, we managed to use optimization with RPE as objective function, target geometry will adjust to the best fit. Though it could be somehow overfit sometimes, it performed as expected when given proper data.

Moreover, a series of trials and experiments to collect good enough image data for calibration and rectification of camera have been carried out through our project. Our final results came out from a time-consuming process when data were collected in steady state each time to also avoid the possible synchronization problem between two systems with extra care to exposure setting, contrast setting and proper focal length. And in particular calibration setting, we also fixed principal point value to be half of the correspond-

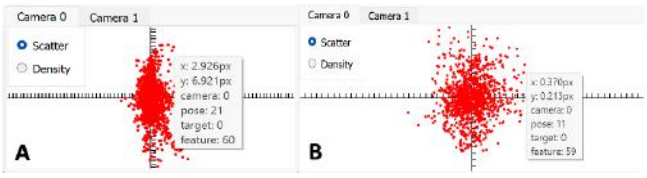

Figure 5. Comparison of Reprojection Error(RPE) with (A) Checker Pattern Dynamic Collection and (B) Charuco Pattern Steady Collection with target geometry optimization

ing resolutions.5 As a result, with optimized calibration process, RMS RPE has achieved down to 0.17 pixel from previously in some cases more than 20 pixels which made it ideal to proceed. Consequently, better result with significantly

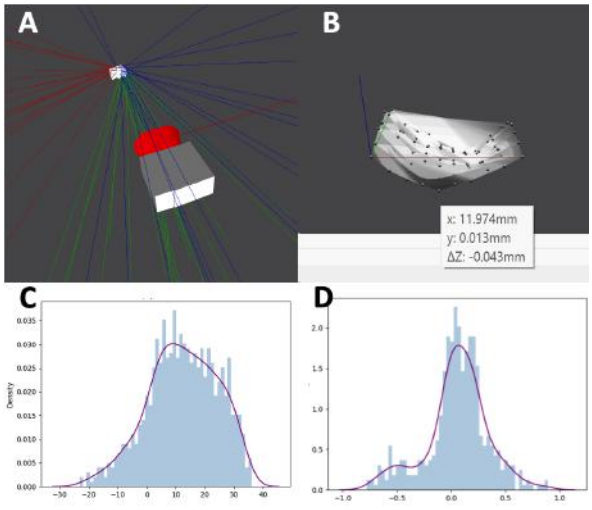

Figure 6. (A) Stereo Cameras illustration after rectification (B) Geometry after optimization which indicated our flaws in target (C) Epipolar Error for Checkerboard target with dynamic collection of data (D) Epipolar Error for Charuco target with steady collection of data

decreased Epipolar error and standard deviation along with it in camera rectification step. From originally 11.4 ± 11.9 pixels to 0.05 ± 0.28 pixel. This will boost the accuracy of pose estimation on target pattern based on the much more accurate intrinsics.6

Having identified the remaining problem that interlaced issue existed in video capture which is one of the crucial factors influenced the dynamic image quality, two directions of methods have been tried. First, it is hardware driver original tuning. Since it is a system encompassing dedicated Sony graphics unit and Blackmagic boards in the desktop driven by Gstreamer which is a common library to transfer images flow. We enabled the option to do deinterlace on board rather than on desktop as an additional computational burden. Second method is to do post processing using mainly vanilla linear interpolation on images since the nature of interlacing is mix odd row and even ones in one image as scanning manner. Time delay between optical tracking system and microscope has been measured to be around the same 34ms by time difference between abrupt change of motion of tracked objects in both systems.

2.4. Hand-eye Calibration

It's essential to get each part within the system accurately tracked, and one method of tracking the microscope is performing a hand-eye calibration. While the Atracsys and microscope are not communicating, the first step should be the synchronization of timestamped data from Atracsys and microscope.

We deployed distributed master-slave communication of multiple computers in ROS environment, using the ROS build-in time-synchronizer to make two sets of sensing data aligned. We also designed unit test to verify the effect of synchronization: We first start tracking the static optical marker within the FOV of microscope, and took a sudden movement of the optical marker, then we find the movement change frame in stereo video, and the sudden change point in tracking data. We get these two timestamp to cacualate the difference, which is $22ms$.

The second step is hand-eye calibration, we set it up as a $AX = XB$ problem 7.

We use different ways to solve it. As the first try, we use the quaternions to represent rotations [4], and solve for rotation, then translation. The drawback is that rotation error propagate to translation, since translation relies on rotation.

We then refer to Furrer et al. [5] They built a toolbox for hand-eye calibration based on SVD by using a dual-quaternion representation for both rotations and translations. This method can simultaneously solve rotation and translation, thus solve the problem of error propagation. Besides, it filter out outliers and noise in pose pairs by screw-axis theory and dual-quaternion scalars, and it uses RANSAC to iteratively optimize hand-eye calibration result. At last perform a joint maximum likelihood optimization as the refinement of calibration. This method potentially improves the hand-eye calibration accuracy.

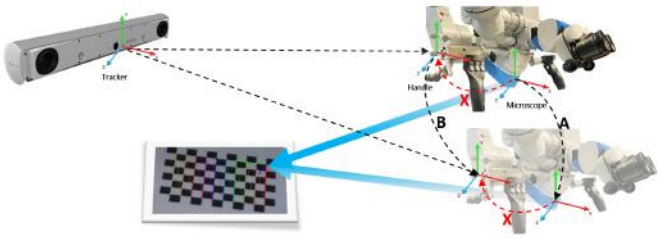

Figure 7. Hand-eye Calibration Setup

We also designed an method to evaluate the overall effect of hand-eye calibration 8. Take the first frame to get the transformation from target to optical tracker, since target is relatively static to the tracker, we can inference the camera pose w.r.t target coordinate for the rest frames.

$$F_{CT_{[0]}}^{*-1} X^{*-1} F_{Oh_{[0]}}^{*-1} F_{Oh_{[t]}}^{*} X^{*} = F_{TC_{[t]}}^{*}$$

F^* means the real transformation with errors.

Then we can reproject the known target 3D coordinates to

the image coordinate, and count the mean RPE(reprojection error) between reproject coordinates and target corner coordinates among all frames 9. The first hand-eye calibration method gave us 20 pixels mean RPE, equivalent to 0.6mm in our setup, while the second method gave us 16 pixels, equivalent to 0.48mm. Obviously the second method is better.

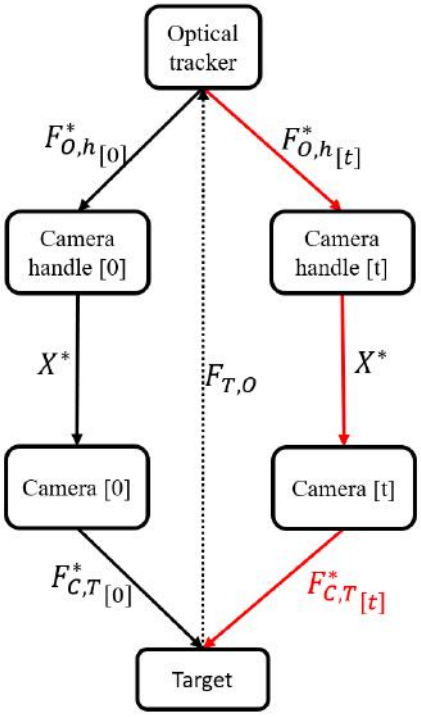

Figure 8. Hand-eye Calibration Evaluation

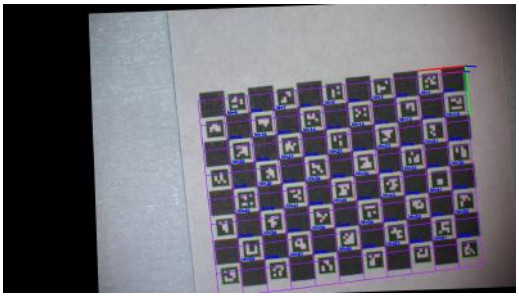

Figure 9. Reprojection result example

2.5. DNN Based Inference

As hand-eye calibration is not the only choice when consider tracking the microscope, we designed another method, which is based on DNN and point cloud registration.

As shown in the setup 10, there are two things we need to know to get the hand-eye transformation, one is the location of phantom in tracker coordinate F_{TPh}, another is the location of phantom in microscope coordinate F_{MPh}.

To get F_{TPh}, we put stereo image into the motion network

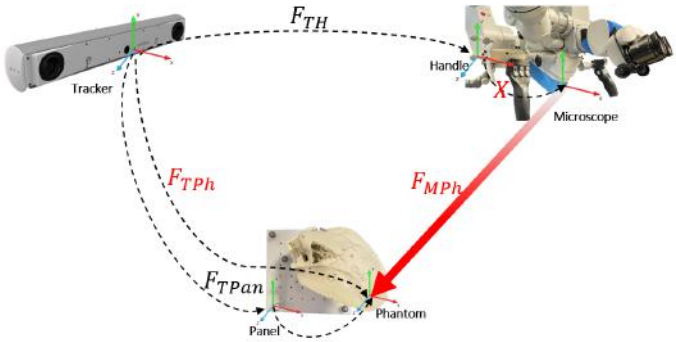

Figure 10. DNN Based Inference Setup

to inference the point-cloud of local phantom surface in microscope coordinate, then we register that surface to the own coordinate of phantom 11.

F_{MPh} requires the phantom point-cloud in tracker coordinate. The easy and low-cost way to acquire the point-cloud is to use a tracked pointer tool to sample points on the phantom surface, and pivot calibration for that pointer is needed.

We tried two method for pivot calibration, the Algebraic One Step Method and a RANSAC method. They are evaluated by the RMS residual, the first method got 0.13mm, while the second got 0.03mm, which is more ideal.

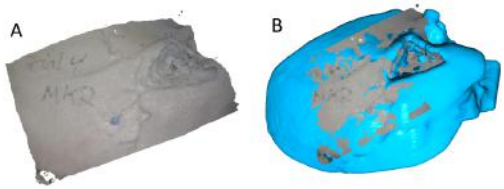

Figure 11. (A) Point-cloud Inferenced from Motion Network and (B) Surface Registration Result

We sampled 280 points on phantom surface by subscribing the position of pivot from ROS each time we touched the phantom. Then ICP algorithm is performed to register the phantom to tracker coordinate 12.

For we fixed the phantom to a panel, which is also tracked by the optical tracker, we can easily track the phantom by tracking the panel.

Now that hand-eye transformation can be calculated by

$$X = F_{TH}^{-1} F_{TPh} F_{MPh}^{-1}$$

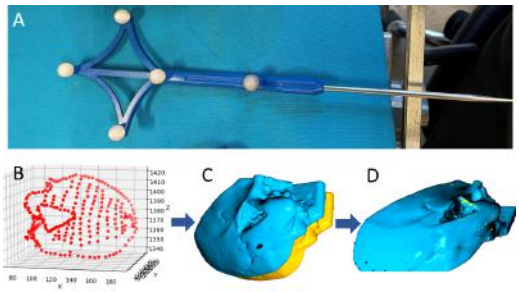

Figure 12. (A) Pointer Tool (B) Sampled Point-cloud (C) Registration Process (D) Registration Result

2.6. Sensitivity Analysis

The navigation system, involving optical tracking and microscope vision, tends to accumulate errors. To quantitatively analyze the errors and determine the decisive part that affects the overall errors, sensitivity analysis is crucial 13.

As a goal-directed analysis, we mainly want to quantify

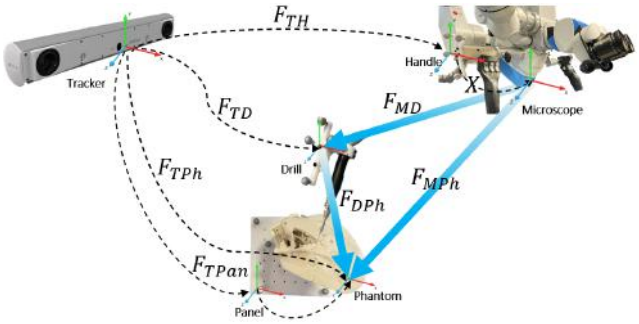

Figure 13. Navigation System Setup

three part of errors: Error between microscope and drill, error between microscope and phantom, and relative error between drill and phantom.

While for intraoperative scene, phantom is relatively stable w.r.t the microscope and tracker, our main focus lies in the drill motion tracking. The expression of real drill pose in microscope coordinate is:

$$(F^*_{TH}X^*)^{-1}F^*_{TD} = F^*_{MD}$$

The rotation error:

$$\vec{\alpha_{MD}} = \begin{bmatrix} I, & -R_{DM}, & -R^{-1}_{TD}R_{TH} \end{bmatrix} \begin{bmatrix} \vec{\alpha_{TD}} \\ \vec{\alpha_X} \\ \vec{\alpha_{TH}} \end{bmatrix}$$

$$\vec{\alpha_{MD}} = \begin{bmatrix} 0.087 \\ 0.087 \\ -0.087 \end{bmatrix} rad$$

The translation error:

$$\vec{\varepsilon_{MD}} = \begin{bmatrix} I \\ -R^{-1}_{MD}R^{-1}_X sk(R^{-1}_{TH}(t_{TH} - t_{TD})) \\ -R^{-1}_{MD} sk(R^{-1}_X R^{-1}_{TH}(t_{TH} - t_{TD}) + R^{-1}_X t_X) \\ -R^{-1}_{MD}R^{-1}_X \\ -R^{-1}_{MD} \end{bmatrix}^T \begin{bmatrix} \vec{\varepsilon_{TD}} \\ \vec{\alpha_{TH}} \\ \vec{\alpha_X} \\ \vec{\varepsilon_{TH}} \\ \vec{\varepsilon_X} \end{bmatrix}$$

$$\vec{\varepsilon_{MD}} = \begin{bmatrix} 0.1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.7 \\ -0.7 \end{bmatrix} + \begin{bmatrix} 0 \\ 12.6 \\ 12.6 \end{bmatrix} + \begin{bmatrix} -0.1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 10.7 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 10.7 \\ 13.3 \\ 11.9 \end{bmatrix} mm$$

From this part, the error of microscope to drill is domained by hand-eye transformation where we denoted as X.

As we need to reproject the drill to image coordinate, we also evaluated the reprojection error:

$$\Delta U = (K\vec{\varepsilon_{MD}} + \Delta K\vec{t_{MD}})/Z$$

Reprojection error determined by microscope to drill translation error and microscope calibration sensitivity.

The relative error between drill and phantom, translation error:

$$\vec{\varepsilon_{DPh}} = \begin{bmatrix} I \\ -R^{-1}_{DPh} sk(R^{-1}_{TD}(t_{TD} - t_{TPh})) \\ -R^{-1}_{DPh} \end{bmatrix}^T \begin{bmatrix} \vec{\varepsilon_{TPh}} \\ \vec{\alpha_{TD}} \\ \vec{\varepsilon_{TD}} \end{bmatrix}$$

Overall, the sensitivity analysis results guided us to perform further optimization on hand-eye calibration, and camera calibration, the limitation of our setup is clearly shown. We'll explore more on relative error, especially from microscope side in the future.

2.7. VR Environment and Data Generation

Aside from setting up proper environment for previous work AMBF+ as described in detail inside the documentations provided in Github where we start to build upon, we also extend the possible usage scenarios by adopting Xserver, VNC, etc. to complement existing docker image. This will enable cross-platform deployment giving more flexibility resulted from different machine, operating system, ROS version. By adopting these new libraries will, in addition, harness the power of GPU better and provide a easy to use port to stream image status in real time using just web browser just like using the actual Ubuntu with correct setting for interactive development and visualization. This feature has been tested on MacOS and Ubuntu 21.10, 20.10 to ensure its functionalities.14 New Plugin module to collect the data stream from optical tracker has been developed based on rospy to subscribe rostopic and publish in AMBF object command interface to interact the virtual objects meanwhile the data produced in VR simulator can be subscribed by either hardware like VR headset or simply recorded from designated topics.

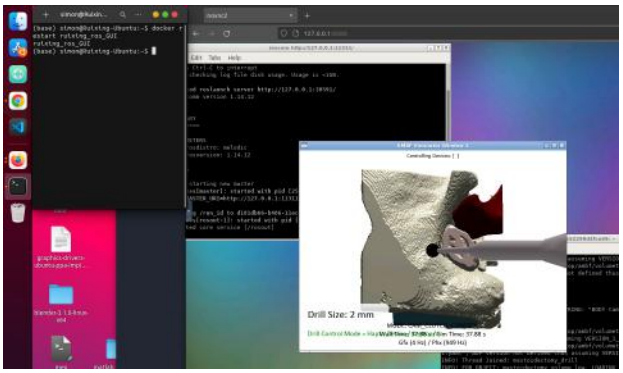

Figure 14. Enhanced VR Docker Image Illustration Web Interactive Environment

3. Experiments

Preliminary Experiments to utilize real images and registration pipelines has been done and submitted in our The Medical Image Computing and Computer Assisted Intervention (MICCAI) paper this year, while the complete virtual world label has not been evaluated due to the time limit. In this section, analysis on results of MICCAI and our design of following wider less constraint setup will be addressed.

3.1. Inference based –Narrow Case

As aforementioned inference method, microscope was designed to be fixed spatially. Only by enforcing this constraint can we retrieve the pose relative to microscopic cameras without hand-eye calibration. Therefore, this was a more of a narrow case to produce reproducible results.

	Synthetic Data							
	$\|\tau_p\|^2$	>1mm	$\|\phi_p\|^2$	>1°	$\|\tau_t\|^2$	>1mm	$\|\phi_t\|^2$	>1°
Keypoint	42.6	85.8%	3.5	47.5%	269.2	99.7%	28.2	82.3%
Scene Flow	1.8	48.2%	0.2	4.3%	27.2	98.1%	3.5	26.1%
ICP	1.5	28.1%	0.1	**0.0%**	20.1	94.2%	2.9	17.1%
Ours	**0.1**	**2.6%**	**0.0**	**0.0%**	**1.7**	**29.9%**	**0.1**	**3.2%**
	Surgical Phantom Data							
	$\|\tau_p\|^2$	>1mm	$\|\phi_p\|^2$	>1°	$\|\tau_t\|^2$	>1mm	$\|\phi_t\|^2$	>1°
Keypoint	7.5	98.8%	0.8	29.6%	102.9	98.8%	16.5	47.5%
Scene Flow	1.4	35.1%	0.2	5.1%	16.3	98.5%	4.1	37.9%
ICP	0.7	19.4%	0.1	0.6%	9.7	98.5%	2.0	23.8%
Ours	**0.2**	**2.1%**	**0.0**	**0.0%**	**4.8**	**91.2%**	**0.7**	**19.6%**

Table 1. Benchmark result on synthetic and surgical phantom data. p, t: patient anatomy and surgical tool. $\|\tau\|^2, \|\phi\|^2$: translation and rotation error. >1mm, >1°: percentage of error larger than given threshold. For all metrics, lower is better. **Bold** indicates best result.

Though we did cross validation on our inferenced point cloud, relative error still exist in ICP algorithm and since the pose is not moving, the cross validation on images only serves as a reference. Let alone the potential error introduced by unrefined calibration of microscope. The best results still showed not only superiority of pose estimation method we proposed but also within certain error registration we have achieved at this stage.

3.2. Hand Eye Calibration –Wide Case

Though data acquired has been mostly evaluated as discussed in previous method section, we have not got the chance to implement and assess this whole process in time. Utilizing highly accurately hand-eye calibration result, spatial constraint we had on microscope could be consequently lifted and make the setting more close to real surgery.

4. Conclusion And Future Work

This paper presents a novel paradigm to learn directly from actual images and virtually generated multi-modal labels without purely rely on virtual reality generated data. Though the effectiveness within the scope of this term have not been comprehensively assessed, the preliminary and module wise test demonstrated its potentials. Specifically, current calibrated and registered pipelines could perform down to 0.6 mm RPE and downstream DNNs performance could also go down to 1mm translation error and 1 ° in rotation error under microscope spatial constraint.

In our future work soon, we shall explore the label generated from the virtual environment robustness by quantitative measurement for instance the Intersection of the union (IOU) score on segmentation mask to confirm labelwisely.Besides, the possibility to realize online optimization on real world noisy data exists since we have virtual world data (Less-noisy) to iteratively refine the actual 3D pose of tracked objects.

Furthermore, the effectiveness could also be examined by other downstream tasks, more general plugin design and implementation could be forseen in near future incorporated in AMBF library. Finally, the significance of our real-data supplemented VR simulator will also be examined for surgeon perspective.

5. Management Summary

5.1. Divide of work

In summary, we collaborate closely on setting up and calibrating both optical tracking system and microscopic stereo cameras and collected data for hand eye calibration as a team. In the second half of semester, Hongchao worked primarily on sensitivity analysis and refinement on hand-eye calibration meanwhile Ruixing worked on VR simulator and Refining the camera calibration in parallel.

5.2. Milestones

In our original plan, we need to accomplish hardware setup by Jan 31st. Initial pipeline including stereo calibration, rectification, de-interlace, and DNN based inference need to be accomplished on Feb 14th. Error analysis and VR integration were expected to be done by Apr 10th, and data generation should be finished by Apr 24th. While in actual, we accomplished hardware setup and initial pipeline on time. There was some delay on VR integration which we accomplished by Apr 25th. For data generation, we generate our first stage data using DNN based inference by Mar 15th, while the second stage data using hand-eye calibration is still in process, we planed to get it done by May 15th. Overall, we accomplished our expected goal, which includes registration pipelines with optimal error and integration with VR.

5.3. Next Steps

We will continue most of our unfinished work in summer. We will Consecutively try to accomplish tasks like we mentioned in future works. But more specifically speaking, we will first reevaluate the existing RPE in the scenario where surgical tool tip is actually used rather than Charuco target. Then we will proceed to finish the second experiment that is to feed data and corresponding label to the DNN with possible finetuning step to assess the current pipelines. And finally we may add the optimization unit to refine the pose as we addressed before.

5.4. Personal Take Away

5.4.1 Ruixing Liang

Besides the knowledge I got to learn out of this project, I also learnt the importance of calibration of camera and became critical of existing paradigm. For industrial skill set, I learnt to communicate properly within group with supervisors and VR SDE beginner skill. And most importantly, I realized the power of teamwork!

5.4.2 Hongchao Shu

There's no doubt that this project is solid with very detailed computer vision and SLAM knowledge. More than all the technical things, it gives me a chance to explore my personal research style, which lays the foundation for my future research. And most importantly, I realized the power of teamwork!

References

[1] Amir Atapour-Abarghouei and Toby P. Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. pages 2800–2810. 1

[2] Antonio Bernardo. Virtual reality and simulation in neurosurgical training. 106:1015–1029. 1

[3] Zhong Chen, Huiyang Liao, and Xianmin Zhang. Telecentric stereo micro-vision system: Calibration method and experiments. 57:82–92. 3

[4] Jack C. K. Chou and M. Kamel. Finding the position and orientation of a sensor on a robot manipulator using quaternions. *The International Journal of Robotics Research*, 10(3):240–254, 1991. 4

[5] Fadri Furrer, Marius Fehr, Tonci Novkovic, Hannes Sommer, Igor Gilitschenski, and Roland Siegwart. *Evaluation of Combined Time-Offset Estimation and Hand-Eye Calibration on Robotic Datasets*. Springer International Publishing, Cham, 2017. 4

[6] Abel J Lungu, Wout Swinkels, Luc Claesen, Puxun Tu, Jan Egger, and Xiaojun Chen. A review on the applications of virtual reality, augmented reality and mixed reality in surgical simulation: an extension to different kinds of surgery. 18(1):47–62. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/17434440.2021.1860750. 1

[7] Sivabalan Manivasagam, Shenlong Wang, Kelvin Wong, Wenyuan Zeng, Mikita Sazanovich, Shuhan Tan, Bin Yang, Wei-Chiu Ma, and Raquel Urtasun. LiDARsim: Realistic LiDAR simulation by leveraging the real world. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11164–11173. ISSN: 2575-7075. 1

[8] Adnan Munawar, Zhaoshuo Li, Punit Kunjam, Nimesh Nagururu, Andy S. Ding, Peter Kazanzides, Thomas Looi, Francis X. Creighton, Russell H. Taylor, and Mathias Unberath. Virtual reality for synergistic surgical training and data generation. pages 1–9. 1, 2

[9] Weichao Qiu, Fangwei Zhong, Yi Zhang, Siyuan Qiao, Zihao Xiao, Tae Soo Kim, and Yizhou Wang. UnrealCV: Virtual worlds for computer vision. In *Proceedings of the 25th ACM international conference on Multimedia*, MM '17, pages 1221–1224. Association for Computing Machinery. 1

[10] Francois Rameau, Jinsun Park, Oleksandr Bailo, and In So Kweon. MC-calib: A generic and robust calibration toolbox for multi-camera systems. 217:103353. 3

[11] Christopher R. Razavi, Paul R. Wilkening, Rui Yin, Samuel R. Barber, Russell H. Taylor, John P. Carey, and Francis X. Creighton. Image-guided mastoidectomy with a cooperatively controlled ENT microsurgery robot. 161(5):852–855. Publisher: SAGE Publications Inc. 2