Learning Feature Descriptors for Pre- and Intra-operative Point Cloud Matching for Laparoscopic Liver Registration

Zixin Yang[1*], Richard Simon[2] and Cristian A. Linte[1,2]

[1]Center for Imaging Science, Rochester Institute of Technology, Rochester, NY, USA.
[2]Biomedical Engineering, Rochester Institute of Technology, Rochester, NY, USA.

*Corresponding author(s). E-mail(s): yy8898@rit.edu;
Contributing authors: rasbme@rit.edu; calbme@rit.edu ;

Abstract

Purpose: In laparoscopic liver surgery (LLS), pre-operative information can be overlaid onto the intra-operative scene by registering a 3D pre-operative model to the intra-operative partial surface reconstructed from the laparoscopic video. To assist with this task, we explore the use of learning-based feature descriptors, which, to our best knowledge, have not been explored for use in laparoscopic liver registration. Furthermore, a dataset to train and evaluate the use of learning-based descriptors does not exist. **Methods:** We present the LiverMatch dataset consisting of 16 pre-operative models and their simulated intra-operative 3D surfaces. We also propose the LiverMatch network designed for this task, which outputs per-point feature descriptors, visibility scores, and matched points. **Results:** We compare the proposed LiverMatch network with a network closest to LiverMatch, and a histogram-based 3D descriptor on the testing split of the LiverMatch dataset, which includes two unseen pre-operative models and 1400 intra-operative surfaces. Results suggest that our LiverMatch network can predict more accurate and dense matches than the other two methods and can be seamlessly integrated with a RANSAC-ICP-based registration algorithm to achieve an accurate initial alignment.

Conclusion: The use of learning-based feature descriptors in LLR is promising, as it can help achieve an accurate initial rigid alignment, which, in turn, serves as an initialization for subsequent non-rigid registration. We will release the dataset and code upon acceptance.

Keywords: Point cloud matching, 3D feature descriptors, Laparoscopic liver registration, Laparoscopic liver surgery, Non-rigid registration.

1 Introduction

In LLS, pre-operative CT or MRI scans offer precise information about vascular and tumor sites. However, during an intervention, it is challenging for the surgeon to mentally fuse the pre-operative images with the intra-operative laparoscopic images. To mitigate this challenge, image guidance systems [1, 2] help surgeons by overlaying the pre-operative information onto the intra-operative scene. One of the crucial components of an image guidance system is the registration, which estimates the transformation between pre- and intra-operative data. In LLR, both 3D-2D [3] and 3D-3D registration [4] methods can be employed; for 3D-3D registration, specifically, the intra-operative 3D surfaces are reconstructed from intra-operative videos [4] and utilized to constrain the registration solutions.

Registration methods can yield rigid or non-rigid alignment. The rigid registration uses manually or automatically detected landmarks to globally align the 3D pre-operative volume data to the intra-operative 3D surface. To better capture soft tissue deformations, non-rigid registration methods are often needed for final alignment. Non-rigid registration techniques entail two fundamental components: surface matching and volumetric model warping. The former identifies a match between the pre-and the intra-operative surfaces. The latter component uses the surface displacements to deform the volumetric model, so that tumor locations or vascular structures identified in the pre-operative model are correctly mapped to the intra-operative scene [2, 5]. The volumetric model can also be used as a constraint in the surface matching or after the surface matching estimation. Non-rigid deformations allow many potential solutions; therefore, constraints are needed to limit solutions. Various constraints have been explored to solve the registration problem, including anatomical landmarks [6], contours [2, 7], as well as biomechanics-based constraints [5, 8].

The use of 3D feature descriptors is beneficial because they can provide automatic initialization and constraints for rigid and non-rigid registration. However, feature descriptors pose several challenges. At first, liver surfaces are very smooth compared to natural scenes, making local features difficult to capture. Second, feature descriptors may not be able to capture global characteristics of the liver because intra-operative data only shows parts of the

liver surface. Furthermore, deformations and surface reconstruction noise may negatively affect extracted features as they distort the shapes.

Several handcrafted features [9, 10] have been studied in liver registration. Although learning-based 3D feature descriptors have been proposed in the computer vision field [11, 12], they are not designed for LLR and, to our best knowledge, have not been applied in LLR. Most learning-based methods assume the scene is rigid [12]; while a few tackle non-rigid cases [11], they assume the principal surface is visible. These two assumptions are often challenged because the liver is globally deformed, and only a small part can be seen in intra-operative data. Pfeiffer [13] *et al.* proposed a learning-based biomechanical model to estimate the displacement field of a volume mesh to an intra-operative point cloud. However, this method requires a coarse alignment, often performed manually. Although several public datasets [4, 6] have been released, there is still no large public dataset or benchmark available to train and evaluate learning-based methods.

This work explores the use of learning-based 3D feature descriptors for 3D-3D LLR though the following contributions: 1. We describe the generation of a large LiverMatch dataset for studying learning-based 3D feature descriptors in LLR. 2. We propose a learning-based 3D feature descriptor network called LiverMatch for 3D-3D laparoscopic liver registration, which uses a Transformer to obtain self-global and cross-global geometry information from super-points while also predicting per-point feature descriptors from the original point clouds. Furthermore, the network also predicts visibility scores, which help the network focus on the visible pre-operative surface. 3. We evaluate the network relative to another network closest to our proposed network and against a traditional registration method on the dataset.

2 Methods

2.1 Problem Setting

We define the point cloud extracted from the surface vertices of a pre-operative liver model as a source point cloud, $\mathbf{S} \in \mathbb{R}^{n\times 3}$. The intra-operative point cloud is referred to as the target point cloud $\mathbf{T} \in \mathbb{R}^{m\times 3}$, and it is assumed to be generated via a stereoscopic video reconstruction, where n and m are the numbers of points, and $n > m$. Hence, the solution to the pre- to intra-operative registration problem is identifying matches between $\mathbf{S}$ and $\mathbf{T}$.

2.2 LiverMatch Dataset

For this work, the source point clouds are generated from 16 liver models from the 3D-IRCADb-01 dataset[1] [14]. The 3D-IRCADb-01 dataset consists of 20 liver models segmented from CT scans. Four liver models (No. 11, 18, 19, and 20) were excluded from this study due to inherent mesh errors. The target, intra-operative point clouds are generated by simulating various deformations

[1] https://www.ircad.fr/research/data-sets/liver-segmentation-3d-ircadb-01/

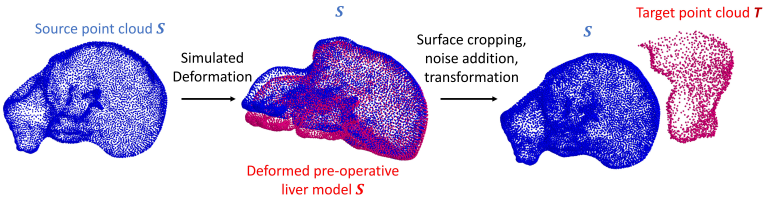

Fig. 1 Schematic description of the generation of the source (S) and target (T) point clouds based on 16 liver surface models from the 3D-IRCADb-01 dataset.

of the 3D pre-operative liver surfaces, then extracting different surface regions following deformation. Fig. 1 illustrates an example of the generation of the **S** and **T** point clouds.

Deformation simulation. We followed the approach described in [13] to generate deformation fields using a neo-Hookean hyperelastic material model with a random Young's modulus (2 kPa to 5 kPa) and a Poisson's ratio of 0.35. We applied up to three forces of 3 N maximum magnitude to random surface regions. In addition, random zero-displacement boundary conditions were also prescribed to areas of radius ranging from 15 - 20 mm. These parameters, along with the CT-derived liver geometry and material properties, were input into a finite element solver, which yielded the deformed models. For this study, we selected the deformed regions featuring a 7–15 mm displacement, mimicking deformations similar to those studied using *in vitro* phantoms in [6].

Target point cloud generation. The following four steps were used to simulate the target point clouds: First, we cropped the front liver surfaces following the simulated deformations and extracted vertices that served as the raw target point cloud. Second, to mimic different visual fields of view of the intra-operative liver surface, we randomly cropped the raw target point clouds using different visibility ratios (m/n). Given a raw target point cloud, a 3D unit direction vector was randomly generated; all vertices in the raw 3D point cloud were projected onto the unit vector, and their distances were ranked. We then selected the top points to ensure a visibility ratio between 0.20 and 0.24, similar to the visibility ratio of 0.22 achieved in the *in vitro* phantom study [6]. Thirdly, random noise was applied on the cropped point clouds with a maximum magnitude of 2 mm, mimicking accuracy levels similar to those achieved by the state-of-the-art stereo matching methods [15]. Lastly, the point clouds were randomly rotated in all directions and translated by up to 20 mm.

2.3 LiverMatch Network

The overview of our LiverMatch network is illustrated in Fig. 2. The network uses the source and target point clouds as input and outputs point-wise feature descriptors, visibility scores, and matches.

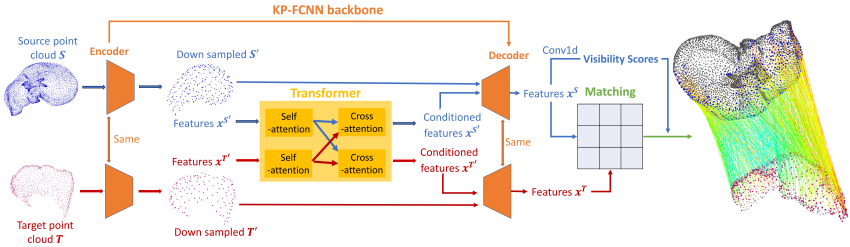

Fig. 2 LiverMatch network overview: 1) Encoder - down-samples input point clouds and extract associated features; 2) Transformer - updates features to conditioned features with self-global geometry and cross-global geometry information. 3) Decoder - up-samples conditioned features to obtain per-point features. 4) Matching - calculates a confidence matrix to select matches. An additional 1D convolution decodes the $\mathbf{x}^{\mathbf{S}}$ to visibility scores.

2.3.1 Encoder

Given $\mathbf{S}$ and $\mathbf{T}$, the encoder extracts super-points $\mathbf{S}'$ and $\mathbf{T}'$ along with their associated features $\mathbf{x}^{\mathbf{S}'}$ and $\mathbf{x}^{\mathbf{T}'}$. In this network, we use the encoder of the kernel point fully convolutional neural network (KP-FCNN) [16]. The encoder consists of ResNet-like blocks and pooling layers based on kernel point convolution (KPConv), which extracts the feature of a point from its neighboring points.

2.3.2 Transformer

The features $\mathbf{x}^{\mathbf{S}'}$ and $\mathbf{x}^{\mathbf{T}'}$ only carry information from their close neighborhood points. To overcome the limitation, the single-head Transformer [17], consisting of a self-attention layer and a cross-attention layer, is applied to update $\mathbf{x}^{\mathbf{S}'}$ and $\mathbf{x}^{\mathbf{T}'}$ with self-global and cross-global geometry information. The self-attention layer allows points from the same point cloud to communicate, while the cross-attention layer allows points from different point clouds to share information. After the transformer, the features will become conditioned features with self- and global-geometry information. Here, we show an example of updating a source feature $\mathbf{x_i}^{\mathbf{S}'} \in \mathbb{R}^{d\times 1}$ using the self-attention and cross-attention layer:

In the self-attention layer, the query vector $\mathbf{q}$, the key vector $\mathbf{k}$, and the value vector $\mathbf{v}$ are first comupted as:

$$\mathbf{q}_i = W_{\mathbf{q}}\mathbf{x}_i^{\mathbf{S}'}, \quad \mathbf{k}_j = W_{\mathbf{k}}\mathbf{x}_j^{\mathbf{S}'}, \quad \mathbf{v}_j = W_{\mathbf{v}}\mathbf{x}_j^{\mathbf{S}'}, \tag{1}$$

where $W_{\mathbf{q}}, W_{\mathbf{k}}, W_{\mathbf{v}} \in \mathbb{R}^{d\times d}$ are learned projection matrices, and $\mathbf{x_j}^{\mathbf{S}'}$ is another source feature. The similarity between $\mathbf{q}$ and $\mathbf{k}$ is measured by:

$$a_{ij} = \text{softmax}(\mathbf{q}_i\mathbf{k}_j^T/\sqrt{d}). \tag{2}$$

$\mathbf{x}_i^{\mathbf{S}'}$ is updated by:

$$\mathbf{x}_i^{\mathbf{S}'} = \mathbf{x}_i^{\mathbf{S}'} + \mathrm{FC}(\mathrm{Concat}[\mathbf{q}_i, \sum_j a_{ij}\mathbf{v}_j]), \tag{3}$$

where $\mathrm{FC}(\cdot)$ denotes a fully connected layer. The same operation is applied to every source feature and target feature.

In the cross-attention layer, $\mathbf{k}$, $\mathbf{v}$ are calculated from the other point cloud. For example, to update the $\mathbf{x_i}^{\mathbf{S}'}$, Eq. 1 becomes:

$$\mathbf{q}_i = W_{\mathbf{q}}\mathbf{x}_i^{\mathbf{S}'}, \quad \mathbf{k}_j = W_{\mathbf{k}}\mathbf{x}_j^{\mathbf{T}'}, \quad \mathbf{v}_j = W_{\mathbf{v}}\mathbf{x}_j^{\mathbf{T}'}, \tag{4}$$

where $\mathbf{x_j}^{\mathbf{T}'}$ is the feature of the target point cloud. The formations with cross attention are the same after replacing the contents for $\mathbf{q}$, $\mathbf{k}$, and $\mathbf{v}$.

2.3.3 Feature Decoding

The conditioned features, along with spatial locations, are then fed to the decoder of KP-FCNN backbone [16] to obtain point-wise feature descriptors $\mathbf{x}^{\mathbf{S}}$ and $\mathbf{x}^{\mathbf{T}}$. Following the decoder, we used a 1D-convolution to decode the $\mathbf{x}^{\mathbf{S}}$ into visibility scores o_v. As only partial points in $\mathbf{S}$ have correspondences, encoding visibility scores helps the network focus on visible points. We clamp the visibility scores within 0 to 1 and create a visibility mask $O_v = [o_v > 0.9]$.

2.3.4 Matching

We first calculate a scoring matrix S and then convert it into a confidence matrix M via the dual-softmax operation [11, 18]:

$$S(i,j) = \mathbf{x}^{\mathbf{S}} \cdot (\mathbf{x}^{\mathbf{T}})^T, \tag{5}$$

$$M(i,j) = \mathrm{Softmax}(\mathcal{S}(i,:)) \cdot \mathrm{Softmax}(\mathcal{S}(:,j)), \tag{6}$$

where $\cdot$ denotes matrix multiplication. Matches are selected from the confidence matrix M via the mutual nearest neighbor criteria: for a pair of matched indexes (i,j), confidence value $\mathcal{M}(i,j)$ should be the maximum value of $\mathcal{S}(i,\cdot)$ and $\mathcal{S}(\cdot,j)$ at the same time. In the end, we use the visibility mask O_v to exclude invisible source points.

2.3.5 Loss Functions

The total loss L of the network is the sum of two loses: $L = L_M + L_v$, where L_M is the matching loss, and L_v is the visibility loss:

Matching Loss. We use the focal loss [19] with the default parameters $\alpha = 0.25$ and $\gamma = 2$ to supervise the confidence matrix M:

$$L_M = -\frac{1}{m} \sum_{(i,j)\in K_{gt}} \alpha(1 - M(i,j))^{\gamma} \log M(i,j), \tag{7}$$

where $\mathcal{K}_{gt}$ is the set of ground-truth matches with same number m as the target point cloud.

Visibility loss. We use the binary cross entropy to supervise the visibility scores o_v:

$$L_v = -\frac{1}{n}\sum_{i=1}^{n} \hat{o}_{v_i} \log(o_{v_i}) + (1 - \hat{o}_{v_i}) \log(1 - o_{v_i}), \tag{8}$$

where $\hat{o}_v$ is the ground truth label, adopting a value of 1 when a source point is visible and 0 otherwise, and n is the number of source points.

3 Experiments

We obtained 700 simulated deformations for each liver model. Our proposed network was tested using the No. 1 and No. 2 liver datasets and trained on the remaining 14 datasets. The training data was generated on the fly using pre-simulated deformations and our intra-operative surface generation methods. We generated one target point cloud for each deformation, hence yielding a total of 9800 deformations for training and 1400 for testing. The model was implemented using PyTorch. We used the SGD optimizer with 35 training epochs and a batch size of 1. Experiments were conducted on a TITAN Xp GPU and an Intel(R) Core(TM) i5-7500 CPU.

3.1 Evaluation Metrics

Given a visible source point $\mathbf{S}(i)$, if the predicted correspondence $\mathbf{T}(j)$ is correct, it will lie within a radius σ from the ground truth correspondence $\mathbf{T}(\hat{j})$, according to [11, 12]:

$$\|\mathbf{T}(\hat{j}) - \mathbf{T}(j))\| < \sigma. \tag{9}$$

Based on the above definition, an inlier ratio (IR) and a match score (MS) can be calculated to evaluate predicted matches, where higher values indicate a better match.

IR calculates the ratio of the number of inliers n_{inlier} to the number of predicted matches n_p:

$$\mathrm{IR} = \frac{n_{inlier}}{n_p}. \tag{10}$$

MS indicates the ratio of the number of inliers to the number of target cloud points m:

$$\mathrm{MS} = \frac{n_{inlier}}{m}. \tag{11}$$

If a registration method is employed to estimate displacement vectors for each source point, the registration error (RE) is measured as the root mean square error between the ground truth displacement vectors $\mathbf{V^{gt}}$ and predicted displacement vectors $\mathbf{V^{pred}}$:

$$\mathrm{RE} = \sqrt{\frac{\sum_i^n \|\mathbf{V}^{gt}(i) - \mathbf{V}^{pred}(i)\|^2}{n}}, \tag{12}$$

where n is the number of source points, $\mathbf{V^{gt}}$ is the sum of deformation and rigid transformation.

3.2 Results

Matching evaluation. We compared our network with the Predator network [12], the closest method to our proposed framework. To adapt the network to deformable scenes, we used ground truth matches to supervise its loss functions instead of the ground truth rigid transformations. The top-k sampling method in Predator was used to select the best candidate source points to match the target points. Finally, the feature descriptors of selected source points and target points were matched with the same matching method we used in LiverMatch.

Table 1 Evaluation of our LiverMatch against Predator network according to the IR and MS evaluation metrics (mean ± std dev.) for different inlier radii σ. *p < 0.05 indicates a statistically significant difference between the LiverMatch and Predator results.

$\sigma(mm)$	0	1	2	3	4	5
			IR%			
*Predator[12]	26.82 ± 4.99	26.94 ± 5.01	27.89 ± 5.10	30.61 ± 5.39	35.36 ± 5.93	41.76 ± 6.60
LiverMatch	37.68 ± 5.91	37.91 ± 5.94	39.58 ± 6.13	43.46 ± 6.61	49.21 ± 7.29	55.69 ± 8.04
			MS%			
Predator[12]	6.90 ± 1.84	6.93 ± 1.84	7.17 ± 1.87	7.85 ± 1.95	9.04 ± 2.11	10.65 ± 2.32
LiverMatch	16.95 ± 3.74	17.05 ± 3.75	17.79 ± 3.83	19.50 ± 4.02	22.04 ± 4.30	24.91 ± 4.64

Table 1 shows the evaluation of our proposed method against the Predator network in terms of IR and MS for a series of inlier radii ranging from 0 to 5 mm. For a 0 inlier radius, both the IR and MS measure exact matches. Nevertheless, with increasing inlier radius, our method yields higher IR and MS values than the Predator network ($p < 0.05$), implying our proposed method can predict denser and more accurate matches than the Predator network.

Ablation study. We conducted an ablation study on the Transformer and the visibility scores of LiverMatch. When the Transformer was replaced with the graph convolution neural net used in Predator, the IR and MS decreased by 4.96% and 2.16%, respectively, for $\sigma = 0$. Furthermore, when the visibility scores were removed from the network, both IR and MS dropped by 6.22% and 3.12%, respectively, also for $\sigma = 0$.

Registration evaluation. We investigated the integration of learning-based point cloud matching with a RANSAC ICP (iterative closest point) rigid registration algorithm. Table 2 summarizes the registration results achieved using the Fast Point Feature Histograms (FPFH) descriptors [20], the learning-based matching point cloud descriptors (Predator and our LiverMatch), and ground truth matches. As FPFH requires heavily down-sampled point clouds, we used

Table 2 Assessment of mean feature extraction time (seconds) and RE(mm) upon integration of descriptors with a RANSAC-ICP registration algorithm. *p < 0.05 indicates a statistically significant improvement in the registration achieved using LiverMatch relative to the other descriptor methods.

	Feature extraction time (s)	Registration method	RE (mm)
*FPFH[20]	0.06	RANSAC+ICP	86.28 ± 49.15
*Predator[12]	0.26		8.88 ± 12.19
LiverMatch	0.07		4.83 ± 3.11
Ground Truth	-		3.89 ± 2.08

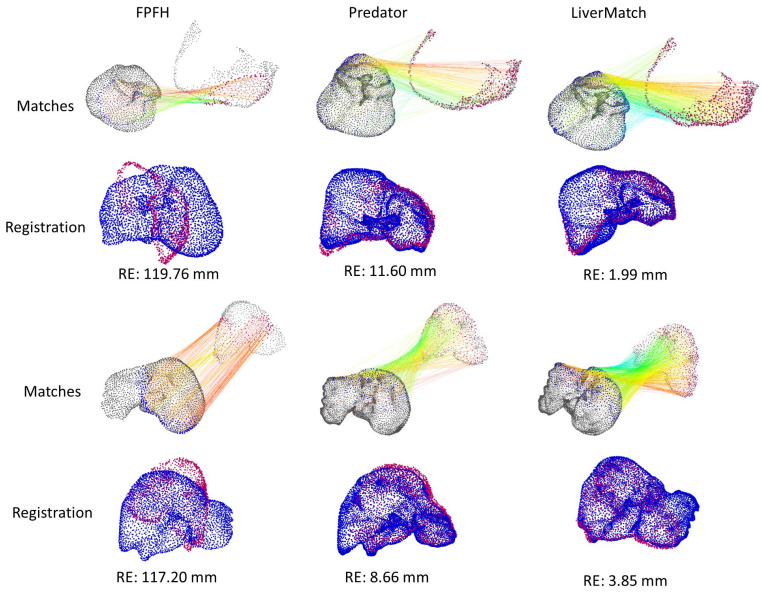

Fig. 3 Visualization of matches and registration results of FPFH, Predator, and LiverMatch on two pairs of the source point cloud (blue) and target point cloud (red). The first two rows show the results for the same pair of source and target point clouds, while the last two show the results for another pair of source and target point clouds. Unmatched points are shown in gray (the first and third rows). Note that point clouds in FPFH are down-sampled.

a voxel size of 5 mm to downsample the source and target point clouds. However, ground truth correspondences were lost after voxelization, so we could not report the IR and MS scores for FPFH. We used the Open3D implementations[2] of RANSAC ICP and FPFH.

As shown in Table 2, the integration of RANSAC-ICP with learning-based matching point cloud descriptors (via Predator and LiverMatch) outperforms the FPFH approach. Moreover, our proposed LiverMatch framework yields the lowest registration error (4.83 ± 3.11 mm), which is comparable to the ground truth registration error of 3.89 ± 2.08 mm and indicates a statistically significant ($p < 0.05$) registration improvement over both Predator and FPFH. The achieved registration results are based on a rigid ICP registration and

[2] http://www.open3d.org/docs/release/tutorial/pipelines/global_registration.html

suggest that a non-rigid registration is needed to further reduce registration error. Lastly, LiverMatch yielded a mean feature extraction time of 0.07 s which was comparable to the mean feature extraction time of FPFH (0.06 s) and much shorter than that of the Predator network (0.27 s).

Fig. 3 illustrates two cases of the point cloud matching and registration results. The target point cloud is occluded in the first case (first two rows). For this challenging case, the learning-based methods can still predict accurate and dense matches. However, FPFH does not yield correct matches, resulting in high registration errors for both cases.

4 Discussion

To generate the LiverMatch dataset, we set limits on the deformation displacements, visibility ratios, and noise magnitude. However, the robustness of learning-based methods to the above factors is still the subject of our ongoing research. Nevertheless, our experiments suggest that our LiverMatch network can find accurate and dense matches between pre- and intra-operative point clouds. Furthermore, the predicted matches can be integrated into rigid-registration methods to achieve fast and accurate rigid alignment.

We tested the Predator and our LiverMatch network on "unseen" liver datasets and their simulated target point clouds. However, the performance of these methods in the clinical setting is still unclear, as it has not been assessed. In addition, whether the learning-based feature descriptors trained on arbitrary objects can generalize to different organs, as speculated in [13], has yet to be further investigated.

On the other hand, Li and Harada [11] proposed a network that includes a Transformer that features a repositioning technique; however, when implementing their approach, the training loss did not converge when the network was trained on our LiverMatch dataset, and, furthermore, the network cannot predict per-point features on original, native resolution point clouds.

We also demonstrated the integration of point cloud matching from learning-based feature descriptors with a rigid ICP registration algorithm. It demonstrated rapid feature extraction and comparable results to ground truth registration. We will further investigate the integration of point cloud matching with a non-rigid registration method. Specifically, we will research a non-rigid registration method that can handle dense matches, outliers, and noisy target surfaces, to more closely mimic typical clinical datasets.

Nevertheless, while acknowledging several limitations and ongoing research efforts discussed here, to our knowledge, this paper constitutes the first investigation of using learning-based feature descriptors for laparoscopic liver registration showing, and it features several promising results, including compelling matching performances, extraction times, and registration results upon integration with rigid ICP.

5 Conclusion

In this paper, we have presented the generation of the LiverMatch dataset to enable us to study the use of learning-based matching descriptors for laparoscopic liver registration, as well as introduced the LiverMatch network that was shown to yield accurate and dense pre- to intra-operative surface matches. Our results suggest that the use of learning-based descriptor matching in conjunction with laparoscopic liver registration is promising, as it not only offers a rapid and accurate rigid alignment of the pre-and intra-operative liver surfaces but also has the potential to assist with non-rigid registration.

References

[1] Haouchine, N., Dequidt, J., *et al.*: Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery. In: 2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 199–208 (2013)

[2] Collins, T., Pizarro, D., *et al.*: Augmented reality guided laparoscopic surgery of the uterus. IEEE Transactions on Medical Imaging **40**(1), 371–380 (2020)

[3] Espinel, Y., Calvet, L., *et al.*: Using multiple images and contours for deformable 3d-2d registration of a preoperative ct in laparoscopic liver surgery. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 657–666 (2021)

[4] Modrzejewski, R., Collins, T., *et al.*: An in vivo porcine dataset and evaluation methodology to measure soft-body laparoscopic liver registration accuracy with an extended algorithm that handles collisions. International journal of computer assisted radiology and surgery **14**(7), 1237–1245 (2019)

[5] Rucker, D.C., Wu, Y., *et al.*: A mechanics-based nonrigid registration method for liver surgery using sparse intraoperative data. IEEE transactions on medical imaging **33**(1), 147–158 (2013)

[6] Suwelack, S., Röhl, S., *et al.*: Physics-based shape matching for intraoperative image guidance. Medical physics **41**(11), 111901 (2014)

[7] Labrunie, M., Ribeiro, M., et al.: Automatic preoperative 3d model registration in laparoscopic liver resection. International Journal of Computer Assisted Radiology and Surgery, 1–8 (2022)

[8] Plantefeve, R., Peterlik, I., *et al.*: Patient-specific biomechanical modeling for guidance during minimally-invasive hepatic surgery. Annals of biomedical engineering **44**(1), 139–153 (2016)

[9] Robu, M.R., Ramalhinho, J., *et al.*: Global rigid registration of ct to video in laparoscopic liver surgery. International Journal of Computer Assisted Radiology and Surgery **13**(6), 947–956 (2018)

[10] Krames, L., Suppa, P., Nahm, W.: Does the 3d feature descriptor impact the registration accuracy in laparoscopic liver surgery? Current Directions in Biomedical Engineering **8**(1), 46–49 (2022)

[11] Li, Y., Harada, T.: Lepard: Learning partial point cloud matching in rigid and deformable scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5554–5564 (2022)

[12] Huang, S., Gojcic, Z., *et al.*: Predator: Registration of 3d point clouds with low overlap. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4267–4276 (2021)

[13] Pfeiffer, M., Riediger, C., *et al.*: Non-rigid volume to surface registration using a data-driven biomechanical model. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 724–734 (2020)

[14] Soler, L., Hostettler, A., et al.: 3d image reconstruction for comparison of algorithm database: A patient specific anatomical and medical image database. IRCAD, Strasbourg, France, Tech. Rep **1**(1) (2010)

[15] Edwards, P.E., Psychogyios, D., *et al.*: Serv-ct: A disparity dataset from cone-beam ct for validation of endoscopic 3d reconstruction. Medical Image Analysis **76**, 102302 (2022)

[16] Thomas, H., Qi, C.R., *et al.*: Kpconv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6411–6420 (2019)

[17] Vaswani, A., Shazeer, N., et al.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[18] Rocco, I., Cimpoi, M., et al.: Neighbourhood consensus networks. Advances in neural information processing systems **31** (2018)

[19] Lin, T.-Y., Goyal, P., *et al.*: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)

[20] Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: 2009 IEEE International Conference on Robotics and Automation, pp. 3212–3217 (2009)