

Supplementary material

Chasing clouds: Differentiable volumetric rasterisation of point clouds as a highly efficient and accurate loss for large-scale deformable 3D registration

Mattias P. Heinrich¹ Alexander Bigalke¹ Christoph Großbröhmer¹ Lasse Hansen²

¹Institute of Medical Informatics, University of Lübeck ²EchoScout GmbH Germany

mattias.heinrich@uni-luebeck.de lasse@echoscout.ai

1. LCSD500

Introducing the “Longitudinal Cancer Screening Dataset” (LCSD500) We release a new point cloud registration dataset based on the publicly available National Lung Screening Trial (NLST) [8, 9], which includes a total of 499 + 15 longitudinal pairs of lung vessel point clouds. For each case (baseline and followup scan), the dataset comprises 4 items:

1. A lung vessel segmentation map
2. A dense vessel keypoint representation
3. A sparse keypoint representation based on density estimation and non-maximum-suppression
4. Corresponding image-based keypoints

For evaluation purposes, we provide 100 expert landmark annotations for a total of 15 cases.

Regulatory information LCSD500 is based upon the publicly available National Lung Screening Trial (NCT00047385) acquired from the Cancer Image Archive (TCIA) [3]. The IRB-approved trial was sponsored by the National Cancer Institute to assess the effectiveness of LDCT compared to chest radiography in reducing deaths from lung cancer. LDCT image data has been released under the *Creative Commons Attribution 4.0 International Public License (CC-BY-4.0)*.

We only perform second-level analysis on anonymized data. Hence, our work and the LCSD500 dataset do not imply additional risks to the participants. Consequently, we also release LCSD500 under *CC-BY-4.0*.

Voxelized lung vessels We collected a large number of paired scans from over 1000 patients of the NLST collection on TCIA¹, for which at least two helical low-dose CT scans

¹<https://wiki.cancerimagingarchive.net/display/NLST/National+Lung+Screening+Trial>

Table 1. LCSD500 dataset overview.

Keypoints	min	max	mean
Dense	25.4k	132.3k	72.6k
Sparse	9.0k	22.0k	11.3k
Corrfield	1.1k	3.6k	2.1k

(baseline with follow up, separated by one year) are available. For this larger corpus, we automatically computed lung segmentations using an in-house segmentation tool and performed a rigid alignment of those binary masks. Based on the Dice overlap, we can estimate the motion magnitude and make a further sub-selection of 550 cases with stronger motion. We resample each image to $224 \times 192 \times 224$ voxels with respect to a uniform voxel spacing of 1.5mm and perform a rigid pre-registration using deedsBCV².

We then employ the open-source, nnUnet-based TotalSegmentator [5, 10] to automatically create lung vessel segmentation maps. We discard all cases with implausible or failed lung vessel segmentations, resulting in a total of 499 scan pairs. To simplify the visualization of pulmonary vessels and to make the synthesis of the point clouds more comprehensible we also publish the generated segmentations as Nifti files.

Vessel point cloud synthesis While sparse keypoints enable the use of algorithms with small memory requirements and can accelerate training and inference of deep learning solutions, dense point clouds are particularly useful for postprocessing and finetuning. Considering the respective advantages, we release generated point clouds in both low (9-11k points) and high (25-72k points) resolutions.

To extract a dense point cloud representation from lung vessel structures, we simply collect every coordinate associated with a vessel segmentation. Starting from this large point cloud representation, we identify a discriminative subset by performing a k nearest neighbour density estimation

²<https://github.com/mattiaspaul/deedsBCV>

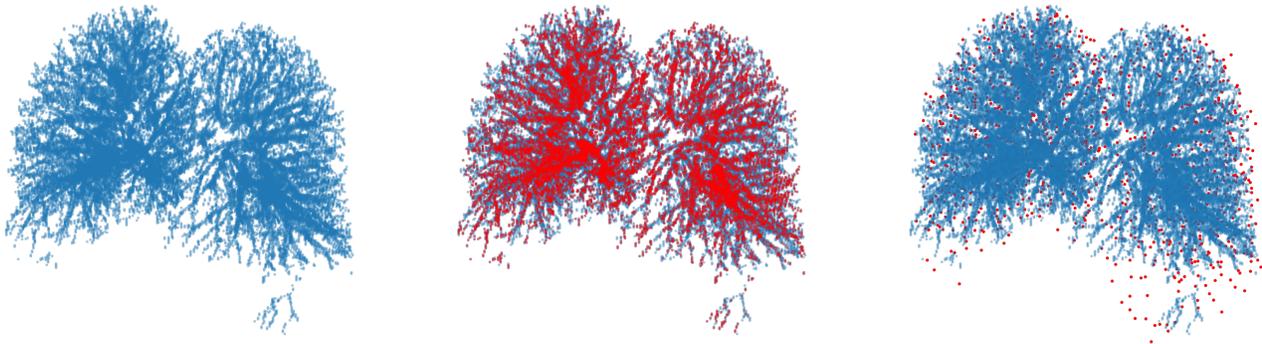


Figure 1. Visual comparison of LSCD500 keypoints. The dense keypoint representation (left, blue) is subsampled into a sparse representation (mid, red). Image-based corrfield keypoints (right, red) do not necessarily match the vessel segmentation.

followed by a non-maximum-suppression. For each point \mathbf{p} in the dense point cloud $\mathbf{P} \in \mathbb{R}^{N_n \times 3}$ we define the calculation of a density score as follows:

$$D(\mathbf{p}) = \frac{1}{k} \sum_{\mathbf{q} \in \mathcal{N}_{\mathbf{p}}^k} (-\exp(-\|\mathbf{p} - \mathbf{q}\|^2 \cdot \sigma^2))$$

depending on the k nearest neighbours $\mathcal{N}_{\mathbf{p}}^k$ and a density weight σ . Then, we obtain the maximum of density estimations from a smaller neighbourhood $\mathcal{N}_{\mathbf{p}}^j$ with $D^*(\mathbf{p}) = \max_{\mathbf{r} \in \mathcal{N}_{\mathbf{p}}^j} (D(\mathbf{r}))$. We add \mathbf{p} to the sparse point cloud if $D(\mathbf{p}) = D^*(\mathbf{p})$, i.e. if it holds the maximum density estimation amongst its j neighbours, yielding a high dispersion. We repeat this process with a slightly incremented σ until at least 9000 keypoints have been chosen.

Corresponding image-based keypoints Although our methods do not rely on external supervision, we provide additional image-derived corresponding keypoints for all cases to facilitate supervised learning by deploying *corrfield*³, a keypoint-based registration framework successfully applied to the task of dense 3D lung registration [4]. *Corrfield* establishes correspondences of Foerstner keypoints by assessing the correlation of gray-value features in baseline and follow-up images using an elaborate discrete optimisation. Fig. 1 and Tab. 1 provide an overview of the published keypoints.

Manual landmarks To evaluate registration algorithms on the LSCD500 dataset, we provide 15 additional cases with expert landmarks.

We collect another set of 100 image pairs (baseline and follow-up) from TCIA and follow the preprocessing described above.

To avoid cases with overly simple registrations, we select only the 15 case pairs with the largest distances between keypoints before and after *Corrfield* registration. A fourth-year medical student established manual correspondences between a total of 100 landmarks for each scan pair (workload approx. 30 hours) using the publicly available software *Isimatch*⁴ [6]. To assess the quality of image-based keypoints in our dataset, we employ the corrfield deformable registration algorithm on the LSCD500 test set. Fig. 2 shows the cumulative distribution of resulting target registration errors (TRE). *Corrfield* demonstrates high agreement with manual landmarks with mean TRE decreasing from 10.21 mm (initial/rigid pre-alignment) to 1.83 mm, highlighting the suitability of its generated keypoints as supervision in learning-based methods. In addition to the results mentioned in the main paper, we ran an additional ablation using Adam instance optimisation with DiVRoC twice (2x Adam), which yields a TRE of 2.43mm and performs on par with selftrain + Adam.

Discussion We anticipate that LSCD500 may serve as an important dataset to bridge the gap between image-based 3D medical image registration cf. [4] and deformable point cloud registration [11]. So far, a direct comparison between these two different 3D alignment strategies has not been readily possible for learning-based methods, because the sizes of 3D lung datasets have been too small (30 cases in [4], 10 cases in [2]). Hence providing a much larger dataset will alleviate such limitations. In addition, our dataset provides very reliable automatic correspondences for this large dataset for supervised learning, which was not done for the PVT1010 data [7].

³<https://grand-challenge.org/algorithms/corrfield/>

⁴<https://www.isi.uu.nl/research/software/isimatch/>

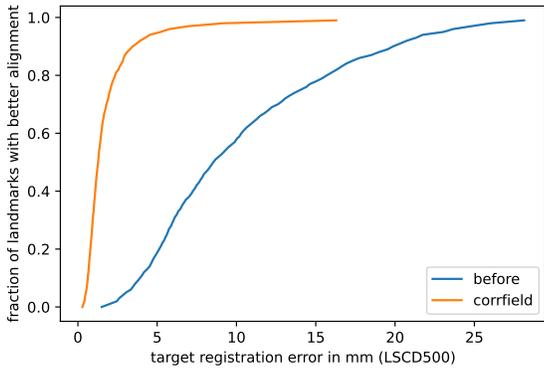


Figure 2. Cumulative distribution of target registration errors (TRE, in mm) of corrfield on LSCD500 dataset.

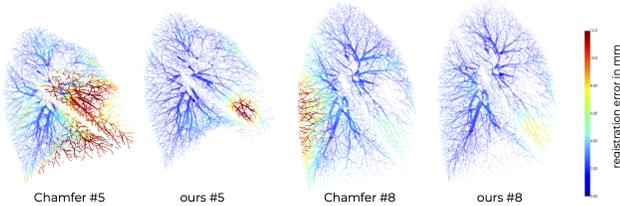


Figure 3. Visualisation of extrapolated TRE on 3D lung geometry.

2. Runtime/complexity and implementation

As described in the main body of our manuscript, the novel differentiable volumetric rasterisation of point clouds (DiVRoC) method provides a substantial gain in runtimes for two important aspects of point cloud registration. First, the distance computation between a displaced source and target cloud can be speed-up by orders of magnitudes compared to the Chamfer distance. Second, the extrapolation and regularisation of a sparse point cloud can also be performed with less memory and much higher computational speed. We analysed the empirical complexity of our method for a range of the main hyper-parameter setting.

In Fig. 4 a comparison of GPU runtimes for 50 iterations of forward and backward paths of the DiVRoC point distance for clouds with a size of 100 thousand points is analysed with respect to the voxel grid size (our choice is either 76^3 or 152^3 in the experimental section) and compared to the Chamfer distance with 16k or 80k points. Our results show a speed-up from 4 secs down to 0.2 secs, despite the fact that we use $6\times$ more points (a number that is too large to even fit into 48GByte of VRAM for Chamfer). Crucially our method is nearly independent of the number of points in the clouds in terms of runtime and memory usage. We strongly believe that this is a very relevant contribution to the field of point cloud and shape analysis. It

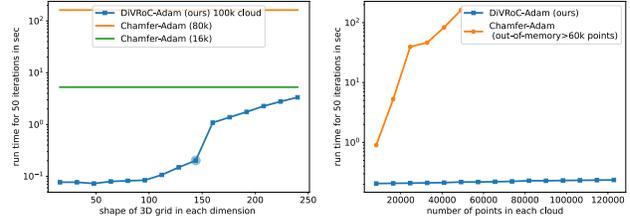


Figure 4. Runtime analysis of differentiable point distances (ours, DiVRoC in blue vs. Chamfer in orange or green) for 50 iterations of optimisation.

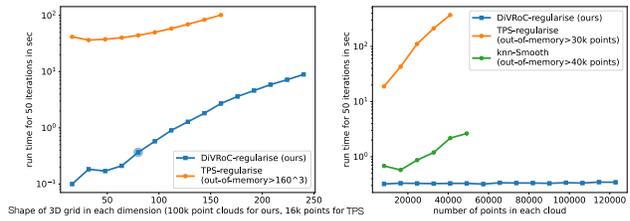


Figure 5. Runtime analysis of the DiVRoC regulariser in comparison to Thin-Plate-Splines and kNN-smoothing demonstrating orders of magnitude faster extrapolation and smoothing of our approach.

Table 2. Ablation for DiVRoC used within DGCNN model comparing effect of grid spacings for distance and regularisation on downstream registration accuracy (TRE in mm).

grid	51^3	62^3	76^3	94^3	114^3	142^3
distance	5.968	3.289	2.781	2.662	2.614	2.736
regular	2.901	2.948	2.781	2.726	2.806	2.892

also enables sub-second instance optimisation on the high-resolution cloud that was not feasible beforehand.

In Fig. 5, the use of DiVRoC as an implicit sparse motion field regulariser is evaluated again with respect to voxel grid size and number of points. In addition to the kNN-smoothing described in the main paper, we include Thin-Plate-Spline (TPS) [1], a popular extrapolation method between irregular and gridded (3D) domains. DiVRoC excels again with (near) constant runtime with respect to point cloud size and very good scalability even for larger 3D grids, whereas TPS quickly runs out-of-memory.

Implementation details In addition to the detailed derivation of the novel loss and its derivatives, we also provide the source code of our pytorch implementation for all parts of our proposed method. Fig. 6 shows the most important excerpt that demonstrates the mathematical reformulation that, together with the automatic differentiation functionality of `autograd.functional.jacobian`, enables us to obtain such high computational efficiency.

Table 3. Per case evaluation of target registration error (in mm) on the 10 DIRlab-COPD cases along with smoothness metrics based on the Jacobian determinant and statistical tests (Wilcoxon rank-sum) for significance of improvements. Bold cases have $p < 0.01$ compared to Chamfer, underlined ones $p < 0.1$ against the DGCNN without self-training.

method	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	avg	std(log(J))	J<0	time(sec)
Chamfer	4.20	6.63	2.08	3.80	3.92	3.74	2.94	3.70	3.95	9.99	4.49±5.64	0.043	0.03%	1.70
2× Adam	1.62	7.42	1.40	2.16	1.72	1.80	1.45	3.47	1.68	6.55	2.93±5.02	0.039	0.01%	0.53
DGCNN	2.42	4.83	1.41	1.68	2.05	2.23	1.49	2.01	1.50	7.43	2.71±4.33	0.038	0.00%	0.30
selftrain	<u>1.72</u>	<u>4.81</u>	<u>1.42</u>	<u>1.75</u>	<u>1.73</u>	<u>1.90</u>	<u>1.51</u>	<u>1.83</u>	<u>1.51</u>	<u>5.69</u>	2.39±3.36	0.038	0.00%	1.02

```

1 import torch
2 import torch.nn as nn
3 import torch.nn.functional as F
4 from torch.autograd import Function
5 from torch.autograd.functional import jacobian
6
7 class DiVRoC(Function):
8     @staticmethod
9     def forward(ctx, input, grid, shape):
10         device = input.device
11         dtype = input.dtype
12
13         output = -jacobian(lambda x: (F.grid_sample(x, grid) -
input).pow(2).mul(0.5).sum(), torch.zeros(shape, dtype=dtype, device=device))
14
15         ctx.save_for_backward(input, grid, output)
16
17         return output
18
19     @staticmethod
20     def backward(ctx, grad_output):
21         input, grid, output = ctx.saved_tensors
22
23         B, C = input.shape[:2]
24         input_dims = input.shape[2:]
25         output_dims = grad_output.shape[2:]
26
27         y = jacobian(lambda x:
F.grid_sample(grad_output.unsqueeze(2).view(B*C, 1, *output_dims), x).mean(),
grad_output.unsqueeze(1).repeat(1, C, *(1)*(len(input_dims)+1))).view(B*C,
*input_dims, len(input_dims))).view(B, C, *input_dims, len(input_dims))
28
29         grad_grid = (input.numel()*input.unsqueeze(-1)*y).sum(1)
30
31         grad_input = F.grid_sample(grad_output, grid)
32
33         return grad_input, grad_grid, None

```

Figure 6. Excerpt from provided source code for our proposed DiVRoC method that highlights the novel differentiable rasterisation function implemented in pytorch.

All source files with utility functions for training and inference of the proposed models as well as trained models are available at: <https://github.com/mattiaspaul/ChasingClouds>.

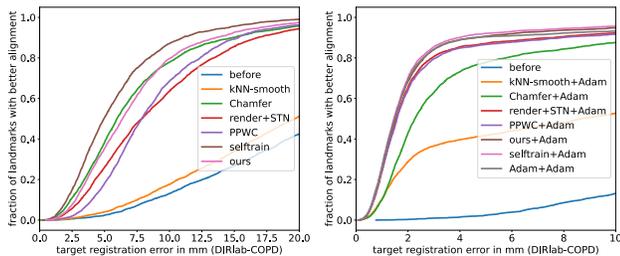


Figure 7. Cumulative distribution of target registration errors (TRE, in mm) of all compared methods on the 10 cases of the COPDgene DIRLAB subset [2].

3. Detailed results and further ablations on PVT1010

Extending the tabular numerical and vector fields visualised on low-resolution point clouds in the main paper, we also show high-resolution renderings of the achieved registration quality of our method (both the DGCNN baseline as well as the self-trained PointPWC-Net) in comparison to the Chamfer-based prior work and the initial misalignment without registration in Fig. 8.

A detailed quantitative case-by-case analysis along with statistical tests and measures of the transformation smoothness based on the Jacobian determinant is provided in Table 3. A smaller standard deviation of the logarithm of the Jacobian values indicates a more plausible displacement field, whereas no negative values ensure a mapping that is invertible and contains no topology-breaking folds. Our two best models achieve the best scores in this category.

Finally, a cumulative distribution plot for all compared ablation experiments with and without Adam instance optimisation is shown in Fig. 7. This highlights the improvement of all three steps in which DiVRoC is employed: regularisation, point distance and instance optimisation.

We performed an additional study on the importance of the hyperparameters that determine the resolution of the grid for distance calculations and regularisation on the quality of the registration compared to our default choice of 76^3 . The results of this ablation for both the DiVRoC distance and regularisation is in Tab. 2. We focus our analysis on the unsupervised DGCNN + Adam, but expect similar hyperparameter settings to hold for PointPWC. For the distance, a further increase in accuracy is possible when enlarging the grid (with little extra compute cost), but the number of grid points should not be reduced by more than a factor of 3, otherwise performance deteriorates. For the regulariser, the TRE is stable over a wide range of hyper-parameter choices with less than 0.18 mm difference across a $20\times$ change in number of grid points.

References

[1] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on*

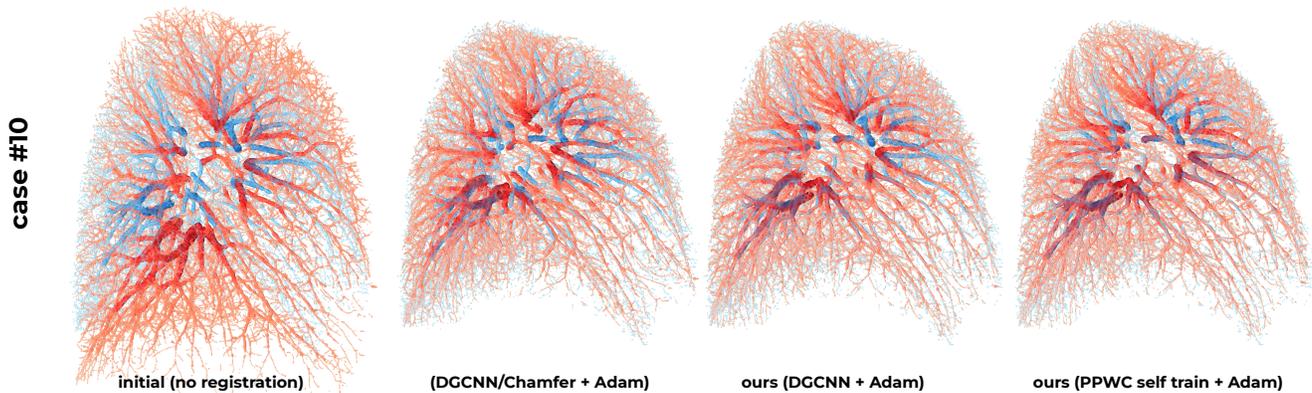


Figure 8. High-resolution point cloud renderings before and after alignment of second exemplary case. Both the large extent of motion that has to be compensated as well as the advantages of our approach compared to the Chamfer distance for training the DGCNN and instance optimisation are visible. Small differences between our best setting (PPWC with self-training) and the simpler DGCNN with DiVRoC can be seen for the visualised Case #10, which is the most challenging for all compared methods. The Chamfer based loss yields unsatisfactory results.

- pattern analysis and machine intelligence*, 11(6):567–585, 1989. 3
- [2] Richard Castillo, Edward Castillo, David Fuentes, Moiz Ahmad, Abbie M Wood, Michelle S Ludwig, and Thomas Guerrero. A reference dataset for deformable image registration spatial accuracy evaluation using the copdgene study archive. *Physics in Medicine & Biology*, 58(9):2861, 2013. 2, 4
- [3] Kenneth Clark, Bruce Vendt, Kirk Smith, John Freymann, Justin Kirby, Paul Koppel, Stephen Moore, Stanley Phillips, David Maffitt, Michael Pringle, Lawrence Tarbox, and Fred Prior. The Cancer Imaging Archive (TCIA): Maintaining and Operating a Public Information Repository. *J. Digit. Imaging*, 26(6):1045–1057, Dec. 2013. 1
- [4] Alessa Hering, Lasse Hansen, Tony CW Mok, Albert CS Chung, Hanna Siebert, Stephanie Häger, Annkristin Lange, Sven Kuckertz, Stefan Heldmann, Wei Shao, et al. Learn2reg: comprehensive multi-task medical image registration challenge, dataset and evaluation in the era of deep learning. *IEEE Transactions on Medical Imaging*, 2022. 2
- [5] Fabian Isensee, Paul F. Jaeger, Simon A. A. Kohl, Jens Petersen, and Klaus H. Maier-Hein. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nat. Methods*, 18:203–211, Feb. 2021. 1
- [6] K. Murphy, B. van Ginneken, S. Klein, M. Staring, B. J. de Hoop, M. A. Viergever, and J. P. W. Pluim. Semi-automatic construction of reference standards for evaluation of image registration. *Med. Image Anal.*, 15(1):71–84, Feb. 2011. 2
- [7] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248, 2017. 2
- [8] The National Lung Screening Trial Research Team. Data from the National Lung Screening Trial (NLST). <https://doi.org/10.7937/TCIA.HMQ8-J677>. 1
- [9] The National Lung Screening Trial Research Team, DR Aberle, AM Adams, CD Berg, WC Black, JD Clapp, RM Fagerstrom, IF Gareen, C Gatsonis, PM Marcus, and JD Sicks. Reduced Lung-Cancer Mortality with Low-Dose Computed Tomographic Screening. *N. Engl. J. Med.*, 365(5):395–409, June 2011. 1
- [10] Jakob Wasserthal, Hanns-Christian Breit, Manfred T. Meyer, Maurice Pradella, Daniel Hinck, Alexander W. Sauter, Tobias Heye, Daniel T. Boll, Joshy Cyriac, Shan Yang, Michael Bach, and Martin Segeroth. TotalSegmentator: Robust segmentation of 104 anatomic structures in CT images. *Radiology: Artificial Intelligence*, 5(5):e230024, 2023. 1
- [11] Zhiyuan Zhang, Yuchao Dai, and Jiadai Sun. Deep learning based point cloud registration: an overview. *Virtual Reality & Intelligent Hardware*, 2(3):222–246, 2020. 2