nature machine intelligence

Article

https://doi.org/10.1038/s42256-025-01025-7

Sparse and transferable three-dimensional dynamic vascular reconstruction for instantaneous diagnosis

Received: 7 July 2023

Accepted: 13 March 2025

Published online: 21 April 2025

Check for updates

Yinheng Zhu[1,2,3], Yong Wang[1], Chunxia Di[4,5], Hanghang Liu[1,2,3], Fangzhou Liao[5] ✉ & Shaohua Ma[1,2,3] ✉

Three-dimensional (3D) structural information of cardiac vessels is crucial for the diagnosis and treatment of cardiovascular disease. In clinical practice, interventionalists have to empirically infer 3D cardiovascular topology from multi-view X-ray angiography images, which is time-consuming and requires extensive experience. Owing to the dynamic nature of heartbeats and sparse-view observations in clinical practice, accurate and efficient reconstruction of 3D cardiovascular structures from X-ray angiography images remains challenging. Here we introduce AutoCAR, a fully automated transfer learning-based algorithm for dynamic 3D cardiovascular reconstruction. AutoCAR comprises three main components: pose domain adaptation, sparse backwards projection and vascular graph optimization. By merging the X-ray angiography imaging parameter statistics of over 1,000 clinical cases into synthetic data generation, and exploiting the intrinsic spatial sparsity of cardiac vessels for computational design, AutoCAR outperforms state-of-the-art methods in both qualitative and quantitative evaluations, enabling dynamic cardiovascular reconstruction in real-world clinical settings. We envision that AutoCAR will facilitate current diagnostic and intervention procedures and pave the way for real-time visual guidance and autonomous catheter navigation in cardiac intervention.

Cardiovascular disease diagnosis and treatment rely heavily on multi-view imaging to probe three-dimensional (3D) cardiovascular structures[1]. For example, in thrombosis therapy, estimating the vessel radius for stenosis severity assessment or distinguishing bifurcations for catheter intervention[2] requires accurate 3D morphological knowledge of coronary vessels. Among various modalities, the most widely used and gold-standard clinical option is two-dimensional (2D) X-ray angiography (XA), because of its high spatial-temporal resolution, deep tissue penetration and the capability of being executed during intervention[3]. Owing to the projective nature of X-ray imaging, 2D XA is always performed from multiple view angles, and the original 3D vascular structure needs to be accurately recovered from these multi-view XA images.

The fundamental challenge arises from the temporal variability inherent in cardiac–respiratory motion. Established algorithms for reconstructing vessels in static body parts, such as head and neck vessels[4], are not suitable for cardiovascular reconstruction. In static body parts, any projection, regardless of the viewing angle

[1]Tsinghua Shenzhen International Graduate School (SIGS), Tsinghua University, Shenzhen, China. [2]Key Laboratory of Industrial Biocatalysis, Ministry of Education, Tsinghua University, Beijing, China. [3]Key Lab of Active Proteins and Peptides Green Biomanufacturing of Guangdong Higher Education Institutes, Tsinghua Shenzhen International Graduate School, Shenzhen, China. [4]Senior Department of Cardiology, The Eighth Medical Center of PLA General Hospital, Beijing, China. [5]ShuKun Technology, Beijing, China. ✉e-mail: lfj@shukun.net; ma.shaohua@sz.tsinghua.edu.cn

https://doi.org/10.1038/s42256-025-01025-7

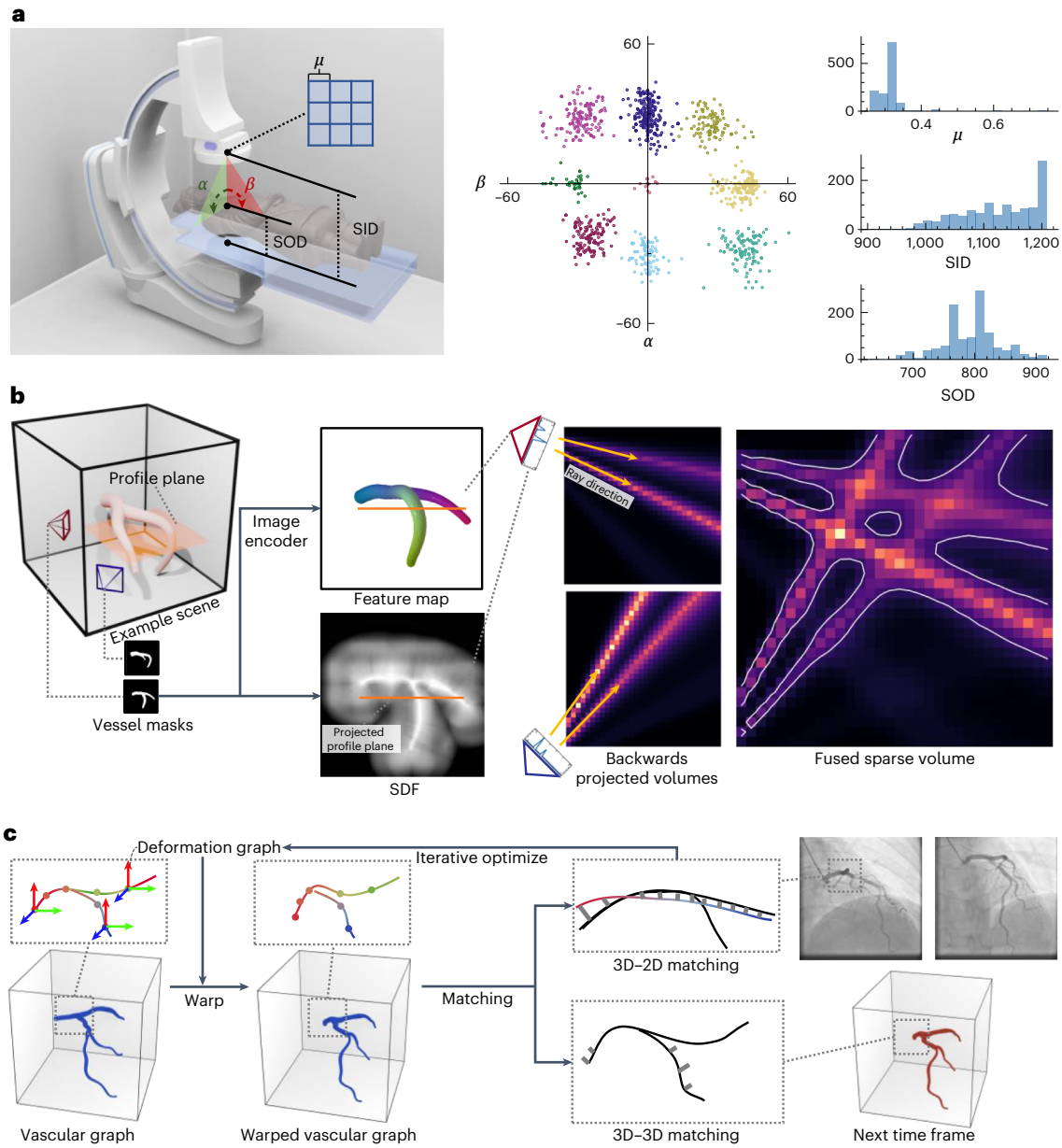


Fig. 1 | Design concept of AutoCAR. a, The rotation angles of the C-arm, denoted by α and β, form a joint distribution represented by nine colour-coded clusters. Other parameters, such as pixel spacing (μ), SID and SOD also follow distinct distribution patterns, as illustrated by the histograms on the far right, where the y axis indicates the count. **b**, Illustration of sparse differentiable back-projection module. The back-projection process from a profile line (orange) in one view (red camera) to the 3D profile plane is shown. Three-dimensional voxels are associated with active pixels, and inactive voxels do not occupy memory or computational resources. **c**, Illustration of graph optimization module. The graph structure is initialized from network prediction and optimized by iterative match-updating procedure.

or acquisition time, captures a consistent 3D structure. However, in dynamic cardiovascular, each XA image represents a transient state. This means that views acquired at different times do not correspond to the same anatomical configuration, breaking the multi-view consistency. Moreover, in clinical practice, cardiovascular XA imaging is typically performed from a limited number of view angles to reduce radiation exposure. Consequently, tomographic methods, relying on dense scanning[5–9], are not directly applicable. The sparse angular sampling, combined with the continuous motion, results in a severely ill-posed inverse problem: reconstructing a temporally evolving 3D anatomical model from a limited set of 2D projections.

Currently, conventional methods for 3D cardiovascular reconstruction face a trade-off between human interaction and accuracy. Specifically, manual annotations of landmarks across multiple images, for example, arbitrary points[10] or segment terminals[11–17], are required in the reconstruction pipeline for satisfactory performance. On the one hand, it has been demonstrated that these manual annotations cannot be well replaced by automated algorithms[18,19]. On the other hand, manual annotations require approximately 200 seconds, making the whole running time of the reconstruction algorithm comparable to or even slightly longer than human interpretation. This limits the effectiveness of providing clinical assistance in real-world intervention. For a detailed survey on related conventional methods, refer to ref. 3.

Recently, deep learning techniques have been widely used on general biomedical reconstruction, including head and neck vascular reconstruction[4]. By utilizing the self-consistency between different projection angles, the relationship between a 3D volume and its 2D

https://doi.org/10.1038/s42256-025-01025-7

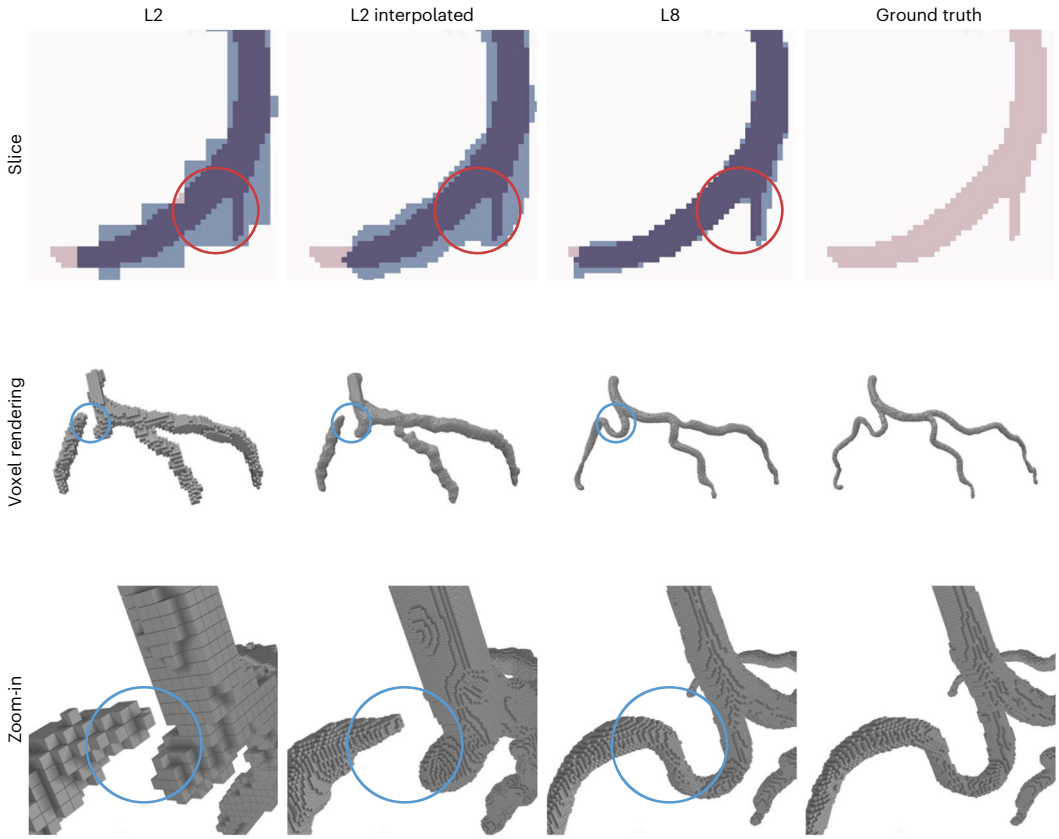


Fig. 2 | Investigation of resolution impacts in sparse and dense models. From left to right, the columns show the L2 prediction (2 mm), interpolated L2 prediction (0.5 mm), sparse L8 prediction (0.5 mm) and ground truth (0.5 mm). From top to bottom, the rows show slice views, voxel renderings and zoom-in details. In the slice views, ground truth (red) and prediction (blue) are overlaid for comparison. Red circles highlight artefacts around vessel boundaries and blue circles indicate broken topology in thin regions.

projections can be effectively learned. Unfortunately, for continuously deforming cardiac vessels, it is infeasible to exploit any self-consistency for self-supervised learning[4], or obtain 3D ground truths for supervised learning[20]. To mitigate this difficulty, two recent studies[21,22] attempted to train deep learning models with synthetic data for 3D cardiovascular reconstruction. As we will show later, their performances diminish seriously when evaluated on real-world XA images.

So far, an accurate, efficient and robust 3D cardiovascular reconstruction solution under the gold-standard XA modality is yet to be developed. In this context, we propose a fully automated cardiovascular dynamic reconstruction method called AutoCAR, which includes the following key designs.

(1) Pose domain adaptation. This utilizes the viewing preferences of XA imaging by human experts, and extends the domain adaptation technique to the pose domain to ensure generalizability in real-world scenarios.
(2) Sparse backwards projection. This leverages the sparsity of vascular structures by treating cardiac vessels as one-dimensional manifolds in 3D space, and ensures detail preservation (fourfold increase in resolution) by restricting computation and storage on the manifold.
(3) Graph optimization. This leverages the curve-network characteristics of vessels for robust vascular topology and temporal coherency.

The AutoCAR designs

The key designs, including (1) pose domain adaptation, (2) sparse backwards projection and (3) graph optimization, are presented here, and details of training (Extended Data Fig. 1) and inference (Extended Data Fig. 2) are provided in Methods.

Pose domain adaptation

Obtaining paired 3D ground truths for supervised training is infeasible, because the cardio-respiratory motions break the multi-view consistency, as shown in Extended Data Fig. 3. Even if electrocardiograms have been used[3,10], multi-view observations are still temporally misaligned. While transfer learning is expected to overcome this difficulty by leveraging synthetic data for training, the statistical differences between real and simulated data often result in limited performance in real-world applications[22]. To address this domain gap, in addition to randomization and adaptation in the image domain[23–25], we extend the domain adaptation to the pose domain by tracking the formation process of real-world XA data.

According to anatomical observations, coronary vessels are generally distributed across mutually perpendicular planes (Figure 3.3 in ref. 26). In clinical practice, experienced interventionalists are able to select specific viewing poses and infer the 3D cardiovascular topology from as few as two views. Through statistical analysis of viewing pose history from 1,215 cases, the preference of viewing poses is confirmed by the clustered distribution of primary angle α and secondary angle β in Fig. 1a. However, other parameters including source intensifier distance (SID), source object distance (SOD) and pixel spacing (μ), are statistically correlated and cannot be casually determined. Thus, instead of uniformly sampling the viewing poses in SO(3), the distribution of real-world viewing poses are exploited and used for synthetic training, to mimic the imaging operations by human experts.

Sparse backwards projection

Sparse backwards projection (SBP) is a differentiable backwards projection neural network module, constructed by sparse submanifold operators[27]. Compared with channel concatenation and dense

Table 1 | Quantitative evaluation in synthetic dataset $\mathcal{D}^{se}$

Category	Configuration ID	Lumen DICE ↑	cl-DICE ↑	CD ↓	HD95 ↓	Resolution
Conventional	Visual hull[36]	0.19	0.26	12.64	33.1	0.5
Learning based	L1[22,28]	0.17	0.17	14.29	32.94	2
	L2[25]	0.49	0.8	3.66	8.75	2
	L3[24]	0.49	0.78	3.87	9.84	2
	L4[4]	0.34	0.40	8.87	25.81	2
AutoCAR (ablation study)	L5	0.48	0.79	3.77	8.77	2
	L6	0.49	0.81	3.68	8.95	2
	L7	0.79	0.92	1.33	3.71	0.5
AutoCAR	L8	0.80	0.93	1.27	3.57	0.5

PMS[40] and PCM[10] are excluded due to their intrinsic non-volumetric prediction. The symbols ↑ and ↓ denote that higher and lower values, respectively, are better. Evaluation metrics include Lumen DICE, cl-DICE, Chamfer distance (CD; mm) and 95% Hausdorff distance (HD95; mm), and 'Resolution' (mm) represents the feature map resolution (voxel size). Details on $\mathcal{D}^{se}$ and the metrics are provided in Methods.

backwards projection, SBP is more memory-efficient while preserving reconstruction details, as detailed below.

Channel concatenation. In previous learning-based methods[22,28], the feature maps from multi-view images are directly channel-wise concatenated and fed into 3D backbone networks, without combining the information of viewing poses into the learning framework. As viewing poses provide natural alignments of multi-view features corresponding to the same 3D position, this direct concatenation strategy leads to limited performance even in synthetic data.

Dense backwards projection. The use of backwards projection in conventional tomographic reconstruction dates back to 1984[5]. In conventional cardiovascular reconstruction methods[3,29–32], it has become a common practice to project, handcrafted and not trainable 2D features, backwards to the 3D space for further process. In addition, conventional methods[29–32] have been proposed to utilize the spatial sparsity of vessels as a regularization term. Learning-based methods[4,24,25] attempt to integrate backwards projections into neural networks, but are all limited to constrained viewing poses, as we will illustrate below. This contradicts the real-world coronary intervention settings with arbitrary free viewing poses. Moreover, when necessary modifications are made to these methods[4,24,25] to support real-world settings, the reconstruction performance suffers from limited resolution, as shown in Fig. 2.

Demand of sparse backwards projection. The thin-curve-network property of vessels requires the neural networks to work at high resolution, whereas projecting 2D feature maps backwards into 3D space consumes a large amount of computation and memory resources. This limits the resolution of feature maps in neural networks. Previous studies have restricted the memory usage for high-resolution feature maps by assuming the scanning trajectory to be orthogonal ($\alpha \in \{0°, -90°, 90°\}, \beta = 0°$)[24] or axial ($\alpha \in [-180°, 180°], \beta = 0°$)[4]. However, these methods are inapplicable for interventional cardiology with the flexible scanning geometry ($\alpha \in \{0°, -90°, 90°\}, \beta \in [-90°, 90°]$).

The SBP module utilizes the sparse nature of vascular structures in both 2D and 3D spaces. More specifically, it restricts the storage and computation only in the vicinity of the vessels, which enables high-resolution and differentiable backwards projection for arbitrary rotational viewing poses. The underlying intuition is that, in both 2D and 3D feature maps, crucial information for reconstruction resides only in the pixels or voxels near the vessels, forming a 1D manifold embedded in 2D or 3D space. Rather than all pixels in 2D feature maps, backwards projecting only relevant pixels on the manifold is more detail-preserving and memory-efficient. This is achieved by formulating backwards projection as well as the reconstruction backbone with submanifold sparse operators[27]. An exemplary scene, consisting of a Y-shaped vessel and two viewing poses, is illustrated in Fig. 1b. For each camera, a vessel mask is obtained by forwards projection during the training phase or 2D vessel segmentation during inference. Given the vessel mask, the 2D signed distance field (SDF) is obtained by the distance transform, while the feature map is obtained by a 2D image encoder. For each pixel with a positive SDF value, 3D voxels along the ray emanating from the pixel are associated, and the corresponding feature vectors and SDF values are assigned. The remaining voxels are considered inactive and do not occupy memory or participate in any computations. After applying such 2D–3D association to individual pixels, backwards-projected sparse volumes can be generated for each view. The backwards-projected voxels from multiple view are then fused according to their active states and SDF values.

Graph optimization

The graph algorithms, designed for the curve-network characteristics of vessels, have been extensively discussed and utilized in the literature[9,33,34]. However, the integration with the dynamic reconstruction system is less explored. The graph optimization module, comprising graph initialization, matching and updating, is specifically designed to bridge the gap between network inference and vascular topology, as well as enhance the temporal coherency.

In more detail, a vascular graph is initialized at each time step, comprising position and radius as node attributes, based on predicted occupancy and centreness from the SBP module. The relative non-rigid transformation among time steps is modelled as a deformation graph[35]. The optimization procedure is illustrated Fig. 1c. In each iteration step, the vascular graph at a specific time step is first warped by the deformation graph, and then matched to the vascular graph and 2D images at the subsequent time step. The obtained correspondences are used to optimize the set of local transformations (rotation matrix R and translation vector $\mathbf{t}$) within the deformation graph, with the objective of minimizing the distance of corresponding points in both two dimensions and three dimensions.

Results

The performance of AutoCAR and existing methods is quantitatively and qualitatively assessed on both synthetic data ('Evaluation in synthetic data') and real-world data ('Evaluation in real-world data'). Further discussions ('Reconstruction for diagnosis and navigation') illustrate AutoCAR's potential to enhance real-world disease diagnosis and therapy.

To ensure fair and feasible comparison, we reimplemented all the learning-based methods in a unified training and evaluation

Table 2 | Quantitative evaluation in real-world dataset $\mathcal{D}^{re}$

Category	Configuration ID	Coverage ↑	EPE ↓	Accuracy ↑
Conventional	Visual hull[36]	0.97	16.4	0.41
	PMS[40]	0.70	6.6	0.48
	PCM[10]	0.68	9.9	0.30
Learning-based	L1[22,28]	0.02	9.0	0.00
	L2[25]	0.60	5.2	0.46
	L3[24]	0.44	2.3	0.36
	L4[4]	0.85	12.2	0.43
AutoCAR (ablation study)	L5	0.64	4.0	0.47
	L6	0.44	2.3	0.36
	L7	0.82	5.0	0.64
AutoCAR	L8	0.92	1.8	0.83

Coverage measures the percentage of predicted correspondences. EPE is expressed in mm. Accuracy is defined as the percentage of correspondences with an EPE below the maximal projected vessel radius (3 mm). Details on $\mathcal{D}^{re}$ and the evaluation metrics are provided in Methods.

framework. This is necessary as existing methods may not release their code[10,36], or the provided code may not be specifically intended for cardiovascular reconstruction in terms of viewing poses[4,24,25], input/output[4,10] and model resolutions[4,22,24,25]. Consequently, the comparison among existing learning-based methods and ablation study of AutoCAR can be regarded as different configurations of a unified model, as shown in Extended Data Table 1. All the methods are trained on $\mathcal{D}^{train}$ defined in 'Mesh and sparse volume preparation' in Methods, with as-fine-as-possible resolution within memory limit, and the same training hyperparameters as in 'Training details' in Methods.

Evaluation in synthetic data

The evaluation in the synthetic dataset $\mathcal{D}^{se}$ ('Mesh and sparse volume preparation' in Methods) is designed for quantitative assessment and analysis of 3D metrics. The underlying reason is that, in the real-world dataset, the paired 3D ground truths are infeasible, thus posing challenges on metrics calculations as well as quantitative analysis. Lumen DICE coefficient, cl-DICE coefficient[37], Chamfer distance and 95% Hausdorff distance are defined and employed as evaluation metrics, corresponding to the reconstruction quality of lumen mask, vessel skeleton, lumen surface (in average level) and lumen surface (in extreme level), respectively ('Evaluation metrics for synthetic dataset' in Methods). The reconstruction is isolated to a single time step. All the metrics are computed at the same resolution of 0.5 mm, and linear interpolation is applied for low-resolution predictions.

Table 1 shows a quantitative comparison of the different methods. Although visual hull[36] is a conventional method based on backwards projection, it shows limited performance because of the ill-posedness of sparse views, demonstrated as 'falsely reconstructed vessels'[10]. The learning-based methods (L1–L4) can alleviate this ill-posedness by learning the data prior of vessels, and outperform the conventional visual-hull method[36]. Among learning-based methods, configurations utilizing backwards projection (L2–L4) outperform channel-wise concatenation (L1). In addition, the adversarial loss, as employed in refs. 22,24, provides limited performance gain, as observed from the comparison between L2 and L5. Notably, with pose domain adaptation and sparse backwards projection, AutoCAR (L8) achieves the best performance compared with existing methods (L1–L4).

As previously mentioned, for static head and neck vascular reconstruction[4], multi-view consistency (projection loss) is used as supervision, which is implemented in L4. In the experiments, adopting projection loss independently leads to limited performance (L2 and L4). When combining projection loss with reconstruction loss as employed in ref. 24, the performance gain remains limited (L2 and L6). This suggests the impracticality of applying static vessel reconstruction methods directly to cardiovascular reconstruction scenarios. The effectiveness of the SBP module can be concluded from the largest performance gain between L2 and L8.

We further investigate the impact of feature map resolution on reconstruction quality, as illustrated in Fig. 2. The predicted dense volumes (L2, 2 mm), interpolated dense volumes (L2 interpolated, 0.5 mm), sparse volume (L8, 0.5 mm) and ground-truth volumes (0.5 mm) are compared in the columns. The slices, voxel rendering and zoom-in details of these volumes are visualized in the rows.

In the slice visualization, L2 predicts coarsely aligned shapes but suffers from artefacts around vessel boundaries (red circle) and broken topology in thin regions (blue circle). These artefacts cannot be removed by interpolation (L2 interpolation, 0.5 mm). The SBP module in AutoCAR (L8) successfully preserves the boundaries and thin-region details by operating feature maps in high resolution.

Evaluation in real-world data

Evaluating the performance on a real-world XA dataset $\mathcal{D}^{re}$ ('Evaluation metrics for real-world dataset' in Methods) offers the most direct assessment of clinical applicability. As 3D ground truths are infeasible for XA imaging, we follow previous practices[3,10] and evaluate the 2D correspondence. More specifically, 2D correspondence refers to the pixel pairs in multi-view images that represent the same 3D location, and is evaluated by projecting reconstructed 3D models to 2D multi-view images.

Specifically, the real-world XA dataset consists of XA videos captured from two views. Considering that most of the compared methods assume a single timestamp, that is, one key frame is selected from the video per view and 2D segmentation is performed to segment the vessels from raw images, we apply the same key-frame selection[38] and 2D segmentation[39] steps to all compared methods where applicable, as detailed in 'Graph optimization' in Methods.

For quantitative evaluation of 2D correspondence (Table 2), we use coverage, end-point error (EPE) and accuracy as metrics, which are detailed in 'Evaluation metrics for real-world dataset' in Methods. Intuitively, coverage is used to measure the percentage of correspondences that can be successfully predicted. EPE denotes Euclidean distances between predicted corresponding pixels and annotated pixels. Accuracy denotes the proportion of correspondences where the EPE is less than the maximum projected vessel radius of 3 mm.

As a conventional backwards projection-based method, visual hull[36] encounters difficulties in effectively filtering falsely reconstructed vessels[10], and is prone to errors in terms of EPE and accuracy. The PMS method[10] has served as a strong baseline in various computer vision benchmarks, including structure from motion and multi-view stereopsis. Despite no vascular- or biomedical-image specific designs, PMS still shows comparable or even superior accuracy relative to conventional methods such as visual hull[36], the PCM method[10] and some learning-based methods (L1–L4). Without backwards projection, neural networks struggle to generalize to real-world data, as evidenced by the comparison between L1 and L2. Without sparse backwards projection, methods L2, L3 and L4 operate on low-resolution feature maps to support flexible scanning geometry and present low reconstruction accuracy. The projection loss (L4 and L6) and adversarial loss (L5) provide limited performance gain, which is consistent with synthetic evaluation.

Besides quantitative indices, we also provide visual comparisons for qualitative assessment, as shown in Fig. 3. We focus on investigating the algorithms dedicated for cardiovascular reconstruction (visual hull[36], PCM[10], L1[22,28] and AutoCAR), while PMS[40] is also included as it gives the second-best quantitative overall score. Figure 4 shows a visual comparison of five representative cases. It can be observed that the deceptive high coverage of the visual-hull method comes at the expense of generating illogical thick output results. PMS produces fragmentary outputs and PCM does not generate satisfactory results without

https://doi.org/10.1038/s42256-025-01025-7

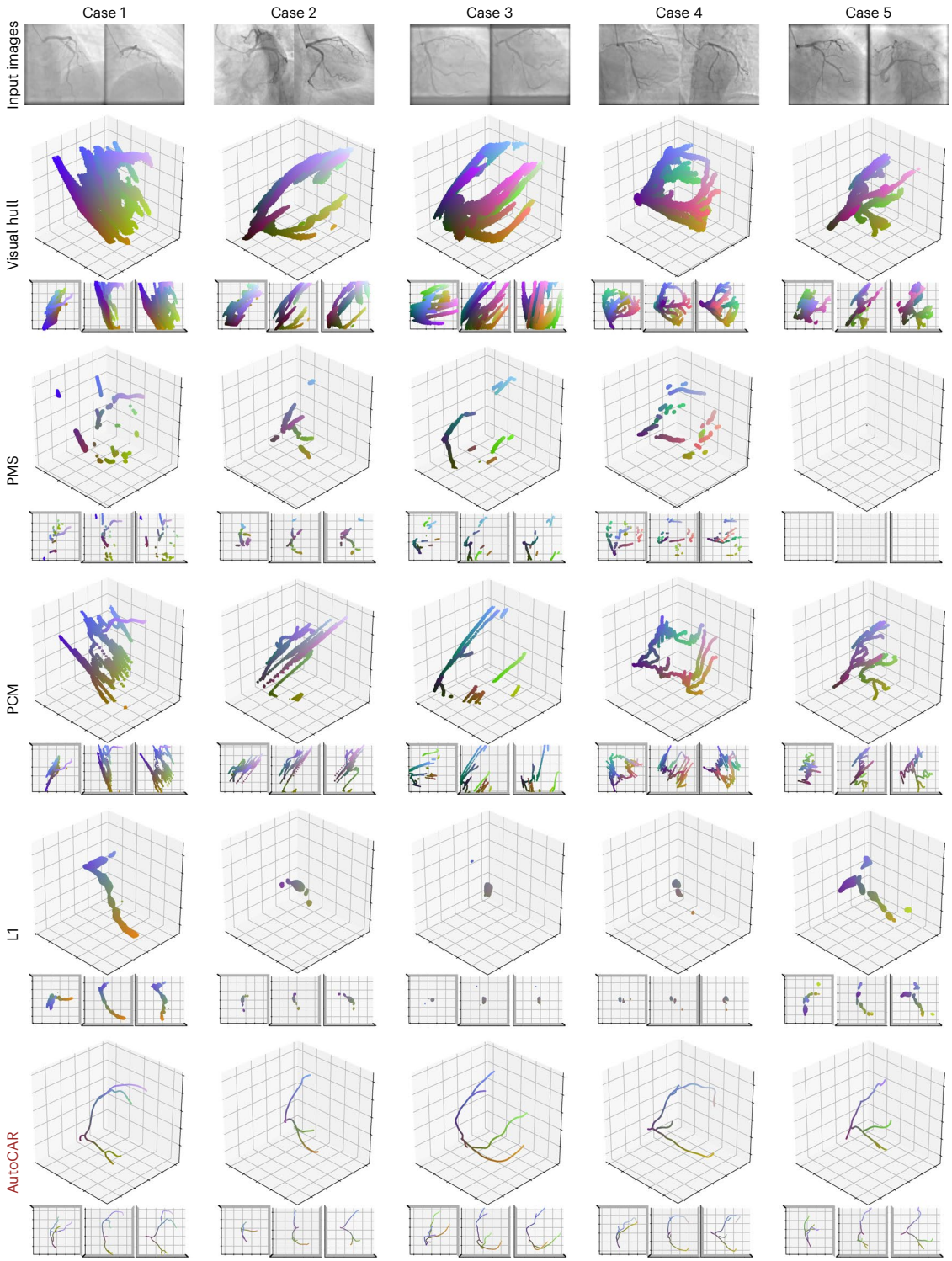


Fig. 3 | Visual comparison between AutoCAR and representative methods. Each column corresponds to an XA imaging case. The top row shows the input images from two scanning views for each case. Subsequent rows show the 3D reconstruction results generated by visual hull, PMS, PCM, L1 and AutoCAR, respectively. Each cube represents a reconstructed 3D volume, with its three-view projections shown below. Interactive and complete visualization can be found at the project website (https://autocar.zyh.science/).

manual steps. Although L1 also adopts synthetic data for training, its performance collapses on real-world images, which can be attributed to both the domain gap between simulation and reality, and the lack of backwards projection in the reconstruction pipeline.

For all cases, our AutoCAR (L8) consistently gives reasonable visual results and outperforms other methods by a large margin in quantitative evaluations at the same time. A detailed comparison of all algorithms across 21 real-world cases, along with interactive dynamic 3D results, is available on the project website (https://autocar.zyh.science/).

Reconstruction for diagnosis and navigation

In this section, we highlight the importance of 3D and multi-view reasoning in clinical tasks during endovascular interventions, such

https://doi.org/10.1038/s42256-025-01025-7

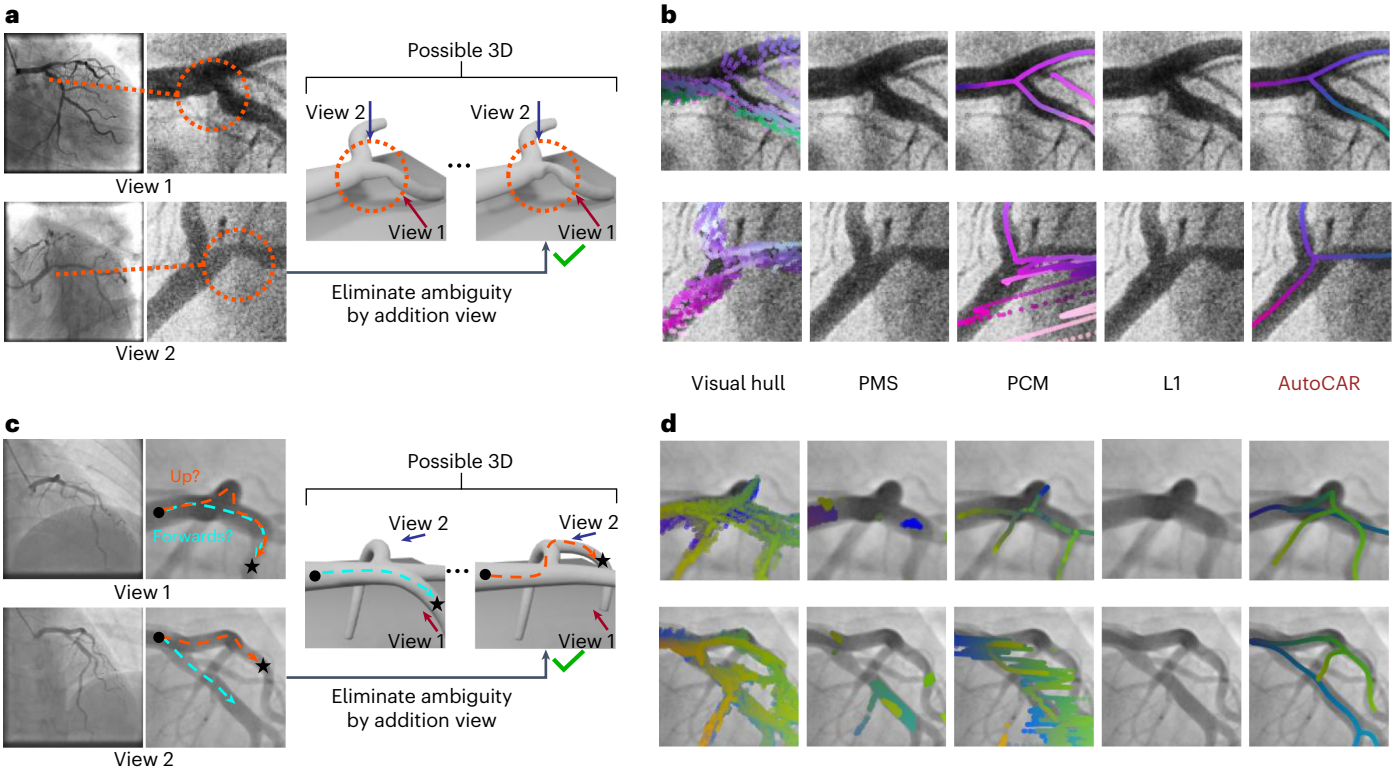


Fig. 4 | AutoCAR provides assistance for cardiovascular diagnosis and endovascular navigation. a, The challenge in 3D cardiovascular interpretation for stenosis diagnosis, presenting two potential shapes from view 1, with view 2 clarifying the ambiguity. **c**, The difficulty of choosing between possible navigation paths due to 3D bifurcation. With the aid of view 2, the correct path pointing upwards (orange) is identified and the wrong path (blue) is excluded. **b**,**d**, Zoom-in illustrations of **a** and **c** respectively, with the reconstruction results of different algorithms overlaid on the original images. The colours indicate spatial locations and are consistent across methods for comparison.

as stenosis diagnosis and navigation. This underscores the necessity of accurate 3D reconstruction and demonstrates the superiority of AutoCAR over existing methods.

In stenosis diagnosis (Fig. 4a), given a single-view observation, assessing stenosis severity is challenging[11] due to distortions from perpendicular overlap and foreshortening effects. Multiple 3D shapes may fit the single image, but they can differ a lot in severity. Therefore, clinicians have to identify the corresponding location in another view and then infer the approximate 3D vessel radius. For endovascular navigation (Fig. 4c), given a single-view observation, it is hard to decide whether to move the guidewire forwards or upwards, considering different interpretations of the 3D structure (3D bifurcations or occlusions). Therefore, clinicians also need to identify the corresponding location in another view and then infer the vascular topology.

The goal of 3D reconstruction is to automate correspondence identification, radius estimation and topology inference steps in the process. Specifically, the reconstruction results of both existing and proposed methods are shown in Fig. 4b,d. These results are obtained under the same input conditions and clinical viewing angles without manual annotations, and with the presence of segmentation noise. The corresponding locations in multiple views are colour-coded. The visual-hull method produces a large number of false-positive outputs under sparse-view conditions. The PMS method[40], however, fails to output results in complex regions. This failure is fundamentally due to its reliance on patch-matching, which struggles in areas with large disparities where different views of the same region exhibit low image similarity. The PCM[10] can extract the topology in the first viewpoint, but it produces large amounts of errors in the second view. This is because the original method requires non-rigid registration based on manual annotations across multiple viewpoints, and its performance degrades under fully automated conditions. In contrast, AutoCAR produces accurate correspondence and topologies aligned with raw images, demonstrating its robustness and reliability in complex clinical scenarios.

In addition to the detailed explanation of the two cases above, the real clinical necessity is evident across all 21 cases we have analysed. In Extended Data Table 2, we provide detailed information on the disease manifestations for these cases, including the locations and severities of the stenoses. When diagnosing these cases, physicians are required to find correspondences between two multi-view videos and to interpret their 3D shapes, similar to the two cases above. With AutoCAR, such correspondences and 3D models can be automatically generated, as shown in Extended Data Fig. 4, encompassing clinically crucial details such as stenoses and bifurcation areas. This serves as robust evidence that AutoCAR can provide assistance for real-world clinical tasks.

Discussion

In this study, we propose AutoCAR, a transfer learning-based solution that achieves fully automated dynamic coronary 3D reconstruction from temporal misaligned sparse-view XA videos. Benefiting from pose domain adaptation, a differentiable sparse backwards projection module and graph optimization, AutoCAR aims to automatize the reconstruction process during navigation, potentially reducing labour-intensive procedures and improving overall efficiency.

The robust and efficient reconstruction of AutoCAR delivers notable improvements across multiple aspects. For instance, AutoCAR achieves a 4-fold increase compared with sparse-view reconstruction methods, and a 1/66 radiation dosage compared with dense-view reconstruction methods. More importantly, AutoCAR yields comparable and robust performance (92% coverage, 1.8 mm EPE and 83% accuracy) to experienced clinicians but is two orders of magnitude faster than human interpretation.

Despite the above advantages, there are a few improvements to be made. First, the inference time is about 7 seconds per time frame, which is mainly occupied by the graph optimization step (6 seconds). Although this time length is acceptable in real-world applications, the running time may be decreased by replacing the iterative matching-optimization with (1) feed-forward inference or (2) using a highly optimized real-time (30 fps) CUDA (a parallel computing platform by NVIDIA) solver, which has been demonstrated in ref. 41. In addition, AutoCAR may fail in cases with inaccurate viewing poses (failure cases (https://autocar.zyh.science/static_hard)), or under conditions of severely degraded 2D segmentation and key-frame selection (Supplementary Note 7). Moreover, it is not applicable to vessels with a radius smaller than 0.5 mm (approximately 2 pixels) due to the presence of weak multi-view consistency and the domain gap between computed tomography angiography (CTA) and XA data. These limitations could be mitigated through collaborative optimization of the scanning process, contrast control and the reconstruction algorithm, which is the future direction of our research.

With AutoCAR, downstream tasks such as navigation or diagnosis can benefit, whether as assistance to humans or as a module integrated into automated robotic systems. In addition, we envision that the design of our proposed domain adaptation strategy will inspire the development of deep learning-based multi-view reconstruction methods in biomedical data processing, where ground truths are typically unavailable. We also anticipate that the design of sparse backwards projection, specifically the introduction of sparse operators, will inspire the development of vascular data processing and analysis, given the inherent sparsity of vascular structures. Finally, we hope that the release of a standardized real-world evaluation dataset annotated by professional interventionalists will respond to the high demand from the community[3,42] and accelerate the development of more powerful methods in this field.

Methods

The training framework follows the self-supervised projection–reconstruction pipeline, as shown in Extended Data Fig. 1. Specifically, before training begins, the triangle mesh and ground-truth sparse volumes, used for forwards projection and loss, are first prepared as detailed in 'Mesh and sparse volume preparation'. During each training step, the multi-view vessel masks are rendered through forwards projection with sampled camera poses from the C-arm pose history, as detailed in 'Forwards projection with pose history'. The masks are then fed into the sparse backwards projection module to obtain the predicted occupancy and centreness, as detailed in 'Sparse backwards projection module'. The sparse DICE and binary cross entropy (DICE-BCE) loss, utilizing predicted and ground-truth sparse volumes, is employed to optimize the SBP module, as detailed in 'Training details'.

During the inference stage as shown in Extended Data Fig. 2, the trained SBP module is combined with graph optimization algorithms to ensure curve-network prior and temporal coherence, as introduced in 'Graph optimization'.

Mesh and sparse volume preparation

Using either CTA volume with modern segmentor or direct annotation volume[43], a volumetric mask can be obtained easily. However, generating a surface triangle mesh and sparse volumes for training requires specific treatment to address the slender nature of vascular structures and accommodate the sparse operator in neural networks.

Vascular structures are slender, with diameters ranging from 0.5 mm to 3 mm. Typically, the voxel size of a mask volume is between 0.4 mm and 0.9 mm. Directly applying the conventional marching cubes algorithm to this volume can result in serious quantization noise and broken artefacts. Therefore, we first upsample the volume to an isotropic resolution of 0.1 mm, then apply Gaussian filtering and finally use the marching cubes algorithm to reconstruct the surface. Considering time and space efficiency (the original volume, with 0.4 mm voxel size and 512^3 shape, leads to the upsampled $2{,}048^3$ volume), the upsampling, Gaussian filtering and marching cubes steps are implemented on sparse data structures. It is important to note that the 0.1 mm resolution is used only for obtaining the surface mesh, denoted as $\mathcal{M}$. Given $\mathcal{M}$, the centreline can be defined as follows and obtained by[44]:

$$\mathbf{p}^{\mathrm{ctl}} = \{p \in \mathbb{R}^3 | \nabla \phi^{\mathcal{M}}(p) = 0, \text{ and } \phi^{\mathcal{M}}(p) > 0\}, \tag{1}$$

where $\phi^{\mathcal{M}}(p)$ is the signed distance function defined as

$$\phi^{\mathcal{M}}(p) = \begin{cases} \inf\limits_{y \in \mathcal{M}} \| p - y \|_2 & \text{if } p \in \mathbb{R}^3 \text{ insides or on } \mathcal{M} \\ -\inf\limits_{y \in \mathcal{M}} \| p - y \|_2 & \text{else} \end{cases}. \tag{2}$$

As the entire neural network is written in a sparse operator, the ground-truth labels used for training, including lumen occupancy and centreness, need to be represented in sparse voxel, with coordinates defined in the world coordinate system.

More specifically, for each 3D mask volume V^{mask} with corresponding volume-to-world transformation, world coordinates of vessel voxels, denoted as $\mathbf{C} \in \mathbb{R}^{n \times 3}$, where n is the number of sparse voxels, are generated by (1) gathering voxel index, (2) applying volume-to-world transformation, and (3) performing sparse nearest-neighbour interpolation. The corresponding label $\mathbf{F} \in \mathbb{R}^{n \times 2}$ consists of two channels for each coordinate: centreness and lumen occupancy. The centreness is defined as

$$F_{j,0} = \frac{\phi^{\mathcal{M}}(C_j))}{\phi^{\mathcal{M}}(C_j) + \inf\limits_{y \in \mathbf{p}^{\mathrm{ctl}}} \| C_j - y \|_2}, \tag{3}$$

where j indexes the jth element among the total n points and $\inf_{y \in \mathbf{p}^{\mathrm{ctl}}} \| C_j - y \|_2$ measures the distance to the centreline.

By definition, it has following properties: (1) if C_j is exactly on the centreline, $F_{j,0} = 1$; (2) if C_j is inside lumen surface, $0 < F_{j,0} < 1$; (3) if C_j is exactly on the lumen surface, $F_{j,0} = 0$. These properties ensure the centreness is normalized to [0, 1] for local vascular structure with any radius, and thus can be used as a soft label to classify whether a voxel belongs to the centreline.

The lumen occupancy, denoted as $F_{j,1}$, can be defined as follows

$$F_{j,1} = \begin{cases} 1 & \phi^{\mathcal{M}}(C_j) > 0 \\ 0 & \text{else} \end{cases} \tag{4}$$

Applying above steps to each mask volume, the prepared data can be expressed as $\mathcal{D}^{\mathrm{CT}} = \{\cdots, (\mathcal{M}^i, \mathbf{C}^i, \mathbf{F}^i), \cdots\}$, where $|\mathcal{D}^{\mathrm{CT}}| = 894$, i represents the index item in the dataset. $|\mathcal{D}^{\mathrm{se}}| = 90$, is randomly sampled from $\mathcal{D}^{\mathrm{CT}}$ and used for synthetic evaluation, while the remaining items, denoted as $\mathcal{D}^{\mathrm{train}}$ are used for training.

Forwards projection with pose history

According to the Digital Imaging and Communications in Medicine (DICOM) standard, viewing pose is defined by primary angle α, secondary angle β, SOD, pixel spacing μ and SID, where α, β and SOD determinate the 4×4 extrinsic matrix T, and μ, SID and SOD correspond to a 3×4 intrinsic matrix K, as illustrated in Fig. 2. Using the multi-view stereopsis convention[45], the projection operation, projecting a 3D point $p \in \mathbb{R}^3$ to pixel index, can be defined as Pp, where $P = KT$. Given collected 1,215 cardiac XA videos, the C-arm pose history is defined as $\mathcal{D}^p = \{\cdots, \boldsymbol{P}^i, \cdots\}$.

Investigating the statistical distribution of these imaging parameters shown in Extended Data Fig. 6, several patterns can be found, including: (1) the joint distribution of α and β are clustered distributed into 9 categories; (2) μ is centred at 0.3 mm with a truncated negative side and a fat tail at the positive side; (3) SID and SOD

are not unimodal and not independent, which suggests that fitting a Gaussian distribution and sampling strategy may not work.

With above definitions, the K-means clustering of viewing pose history $\mathcal{D}^{\mathrm{p}}$ shows nine categories denoted as $\mathcal{D}^{\mathrm{cp}} = \{\{P^i\}_1, \cdots, \{P^i\}_9\}$.

For each training step, two categories $\{P^i\}^{\mathrm{A}}$ and $\{P^i\}^{\mathrm{B}}$ are first uniformly sampled from $\mathcal{D}^{\mathrm{cp}}$ without replacement. Subsequently, one viewing pose is sampled for each category to construct a pose pair, denoted as P^{A}, P^{B}. Together with $\mathcal{M}$, the 2D masks, I^{A} and I^{B}, are rendered using PyTorch3D[46] for each view. To simulate the temporal misalignment, the rendered 2D masks I^{A} and I^{B} are augmented by thin plane spline deformation implemented in Kornia library[47] with parameters of kernel_size=255 and sigma=16. This step occurs online during each training step, enabling data augmentation without the need for pre-computation storage.

Sparse backwards projection module

Given mask image I, the image coordinate near to the mask is regarded as 'active pixels' and defined as:

$$\mathcal{B}^{2\mathrm{D}}(I) = \{(u,v)|\mathrm{EDT}(I)_{u,v} < \epsilon_{\mathrm{v}}\}, \tag{5}$$

where u and v denote the pixel indices in the image, ϵ_{v} denotes the empirical maximal vessel radius and EDT(·) denotes the Euclidean distance transform. For example, $\mathrm{EDT}(I)_{u,v} = 0$ if $I_{u,v} > 0$ and $\mathrm{EDT}(I)_{u,v} = -\phi((u,v))$ if $I_{u,v} = 0$.

The corresponding 2D feature map of I, represented by $f^{\mathrm{hg}}(I)$, can be can be obtained by image feature extractor f^{hg} (ref. 48). The extracted 2D feature map is then derived as follows

$$I^{\mathrm{f}} = \mathrm{EDT}(I) \oplus f^{\mathrm{hg}}(I), \tag{6}$$

where $\oplus$ denotes the channel-wise concatenation operation.

With $\mathcal{B}^{2\mathrm{D}}(I)$ and image-to-world transformation from P, ray bundles in I world coordinates are generated according to ray direction $o + \tau d$ derived from P where o, τ and d refer to origin, sampling length and direction of the ray, respectively. The ray bundles are then further discretized into sparse voxels coordinates $\mathbf{C}^{\mathrm{r}} \in \mathbb{R}^{n\times 3}$ by sampling τ in each ray. The feature vector $\mathbf{F}^{\mathrm{r}}$ for the sparse voxels is then derived as follows.

$$\mathbf{F}^{\mathrm{r}} = \left\{I^{\mathrm{f}}_{x,y} | \forall c \in \mathbf{C}^{\mathrm{r}}, (x,y) = P \cdot (c)\right\} \tag{7}$$

Applying above operation to I^{A} and I^{B}, $\mathbf{C}^{\mathrm{r}}$ and $\mathbf{F}^{\mathrm{r}}$ for each view can be obtained. Then the union and intersection operations on sparse voxels are adopted to fusing features from view A and view B. More specifically, the fused sparse voxels are defined as:

$$\mathbf{C}^{\mathrm{r,A,B}} = \left\{c | \forall c \in (\mathbf{C}^{\mathrm{r,A}} \cap \mathbf{C}^{\mathrm{r,B}}), \mathbf{F}_0^{\mathrm{r,A},c} < \epsilon_{\mathrm{v}}, \mathbf{F}_0^{\mathrm{r,B},c} < \epsilon_{\mathrm{v}}\right\}, \tag{8}$$

$$\mathbf{F}^{\mathrm{r,A,B}} = \{\mathbf{F}^{\mathrm{r,A},c} \oplus \mathbf{F}^{\mathrm{r,B},c} | \forall c \in \mathbf{C}^{\mathrm{r,A}} \cap \mathbf{C}^{\mathrm{r,B}}\}, \tag{9}$$

where $\mathbf{F}^{\mathrm{r,A},c}$ and $\mathbf{F}^{\mathrm{r,B},c}$ denote the corresponding feature vector of coordinates c.

With sparse backbone network[49], denoted as f^{scn}, the reconstruction network is represented as:

$$\begin{aligned}\mathbf{C}^{\mathrm{pred}}, \mathbf{F}^{\mathrm{pred}} &= f^{\mathrm{recon}}(I^{\mathrm{A}}, I^{\mathrm{B}}, P^{\mathrm{A}}, P^{\mathrm{B}}) \\ &= f^{\mathrm{scn}}(\mathbf{C}^{\mathrm{r,A,B}}, \mathbf{F}^{\mathrm{r,A,B}})\end{aligned} \tag{10}$$

Training details

Loss function. Given a pair of predictions ($\mathbf{C}^{\mathrm{pred}}$, $\mathbf{F}^{\mathrm{pred}}$) and labels ($\mathbf{C}^{\mathrm{gt}}$, $\mathbf{F}^{\mathrm{gt}}$) $\in \mathcal{D}^{\mathrm{train}}$, the sparse interpolation[49] is used to convert prediction $\mathbf{F}^{\mathrm{pred}}$ originally defined on $\mathbf{C}^{\mathrm{pred}}$ coordinates to $\mathbf{C}^{\mathrm{gt}}$ coordinates. The interpolated prediction is denoted as ($\mathbf{C}^{\mathrm{gt}}$, $\mathbf{F}^{\mathrm{ipred}}$). To optimize the parameters in network f^{hg} and f^{scn}, a equally weighted DICE and binary cross entropy loss is used and denoted as $\mathcal{L}$.

$$\mathcal{L} = \mathcal{L}_{\mathrm{DICE}}(\mathbf{F}^{\mathrm{ipred}}, \mathbf{F}^{\mathrm{gt}}) + \mathcal{L}_{\mathrm{BCE}}(\mathbf{F}^{\mathrm{ipred}}, \mathbf{F}^{\mathrm{gt}}) \tag{11}$$

It is worth mentioning that as the SBP module is differentiable, the gradient of $\mathcal{L}$ during back propagation can pass through f^{scn} as well as the SBP layer to optimize the image feature extractor f^{hg}, which ensures that the network is end-to-end trainable.

Network architecture. The hourglass network[48] is used as the 2D image encoder, which takes vessel masks as input and outputs 12-channel feature maps. The MinkUnet34C[49] is used as the sparse backbone network. The model is trained using an ADMM optimizer[46] and synchronized batch normalization on 4 Nvidia A10 graphics processing units, with parameters set to batch_size=8, accumulation_step=4 and learning_rate=3e-4.

Graph optimization

In the real-world inference shown in Extended Data Fig. 2, the input comprises two X-ray image sequences captured from different viewing poses. The developed 2D vessel segmentation and key-frame selection techniques are employed as described below. We utilize a pretrained 2D vessel segmentation model based on ref. 39 to convert the sequence of XA images into a sequence of 2D vessel masks. Subsequently, key frames are selected using the method described in ref. 38. It is worth mentioning that 2D vessel segmentation is a well-studied field, and the key-frame selection step can be omitted once an electrocardiogram is available for rough temporal alignment.

For each time step i, the trained reconstruction network predicts the sparse volume $\mathbf{C}^{\mathrm{pred},i}$, $\mathbf{F}^{\mathrm{pred},i} = f^{\mathrm{recon}}(I^{\mathrm{A},i}, I^{\mathrm{B},i}, P^{\mathrm{A},i}, P^{\mathrm{B},i})$. Then, graph optimization is applied to build the vascular graph ('Graph initialization') and deformation ('Graph matching' and 'Graph updating') among the vasculature at different time steps.

Vascular graph and deformation graph. The vascular structure at a single time step is modelled by a vascular graph $G = (N, E, \mathbf{p}^{\mathrm{g}}, \mathbf{r}^{\mathrm{g}})$, consisting of the node set $N \subset \mathbb{Z}$, edge set $E = \{(u,v)|u,v \in N\}$, node position $\mathbf{p}^{\mathrm{g}} \in \mathbb{R}^{|N|\times 3}$ and node radius $\mathbf{r}^{\mathrm{g}} \in \mathbb{R}^{|N|}$.

The deformation graph $G^{\mathrm{d}} = (N^{\mathrm{d}}, E^{\mathrm{d}}, \mathbf{p}^{\mathrm{d}}, T^{\mathrm{d}})$ defines the trainable non-rigid transformation between any pair of time steps, where $T_i^{\mathrm{d}} \in \mathrm{SE}(3)$ are trainable parameters. To adapt the deformation graph into tensor operations in PyTorch, the following modification is made compared with ref. 35:

The warping operation of G^{d} for a point $p \in \mathbb{R}^3$ is defined as

$$\mathrm{warp}(p) = \sum_i^{|N^{\mathrm{d}}|} w_i \left(\cdot T_i^{\mathrm{d}} \cdot \left(p - \mathbf{p}_i^{\mathrm{d}}\right) + \mathbf{p}_i^{\mathrm{d}}\right), \tag{12}$$

where $w_i = \frac{1}{Z} \cdot \exp(-||p - \mathbf{p}_i^{\mathrm{d}}||/2\sigma^2)$ and $Z = \sum_i^{|N^{\mathrm{d}}|} \exp(-||p - \mathbf{p}_i^{\mathrm{d}}||/2\sigma^2)$.

The regularization term is defined as

$$\sum_{(u,v)\in E^{\mathrm{d}}} ||T_v^{\mathrm{d}} \cdot \left(\mathbf{p}_u^{\mathrm{d}} - \mathbf{p}_v^{\mathrm{d}}\right) + \mathbf{p}_v^{\mathrm{d}} - \mathbf{p}_u^{\mathrm{d}} - \left(T_u^{\mathrm{d}} \cdot \mathbf{0}\right)||_2. \tag{13}$$

Graph initialization. In the initialization step, the vascular graph and deformation graph are derived from predicted centreness and lumen occupancy for each time step.

For the vascular graph, given the network prediction $\mathbf{C}^{\mathrm{pred}}$, $\mathbf{F}^{\mathrm{pred}}$, G, the graph node $\mathbf{p}^{\mathrm{g}}$ is first obtained by morphological thinning on the sparse volume. The edge set is built by searching for the nearest neighbours within a radius, followed by constructing a minimal spanning tree (MST), written as

$$E = \mathrm{MST}\left(\left\{(u,v)|\forall u,v \in \mathbb{Z} \cap [0, N-1], ||\mathbf{p}_u^{\mathrm{g}} - \mathbf{p}_v^{\mathrm{g}}||_2 < \epsilon_{\mathrm{v}}\right\}\right). \tag{14}$$

For any pair of connected components of G, denoted as G^i and G^j, the connecting path is defined as a sequence of points starting from any point in G^i and ending at any point in G^j. The cost of a connecting path is the sum of $\exp(-\mathbf{F}^{\text{pred}}_{\arg\min||\mathbf{c}^{\text{pred}}_n - p||_2, 0})$ at each point p on the path. Then, the minimal cost connecting path is the connecting path with the minimal path cost, which can be obtained efficiently by volumetric Dijkstra's algorithm. The edge set E is then updated by (1) finding a pair of G^i and G^j that minimize connecting cost and (2) updating E by the path that connects G^i and G^j. The iteration stops when there is only one connected component or the threshold is exceeded.

For the deformation graph, $\mathbf{p}^{\text{d}}$ is sampled from $\mathbf{p}^{\text{g}}$ by

$$\mathbf{p}^{\text{d}} = \left\{10 \times \left\lceil \frac{p}{10 \times \epsilon_{\text{v}}} \right\rceil | \forall p \in \mathbf{p}^{\text{g}}\right\}, \tag{15}$$

where E^{d} is built with k-nearest neighbours (KNNs), given by

$$E^{\text{d}} = \{(u, v) | u, v \in \mathbf{p}^{\text{d}}, u \neq v, v \in \text{KNN}(u)\}. \tag{16}$$

Graph matching. The graph-matching algorithm is designed to obtain the correspondence between a vascular graph G and a set of points $\mathbf{p}^{\text{s}}$, which is further used in graph updating and radius retrieval. This is achieved by decomposing G into a set of curve segments, which is then converted into a curve-scatter matching problem.

More specifically, trivial nodes of G are nodes with degree 2, and a curve segment is a path in G consisting only of trivial nodes, which can be obtained by iterative edge collapsing. The curve segment set is denoted as $\{\cdots, \text{path}^i \subset N, \cdots\}$, and the position of points in path^i is $\{\mathbf{p}^{\text{g}}_n | \forall n \in \text{path}^i\}$. The path^i–$\mathbf{p}^{\text{s}}$ correspondence is then obtained by dynamic programming as detailed in Algorithm 1, and then assembled into the G–$\mathbf{p}^{\text{s}}$ correspondence.

Algorithm 1 Curve-scatter matching by dynamic programming.

Require $\mathbf{p} \in \mathbb{R}^{n\times 3}$, $\mathbf{q} \in \mathbb{R}^{m\times 3}$. ▷ $\mathbf{p}$ are the curve points, and $\mathbf{q}$ are the scatter points

Ensure $\hat{\mathbf{p}} \in \mathbb{R}^{n\times 3}$ ▷ $\hat{\mathbf{p}}$ are the matched curve points in $\mathbf{q}$.

$\mathbf{f} \Leftarrow \{\text{inf}\}^{n\times m}$ ▷ $\mathbf{f}[i][j]$ is the minimal cost of matching $\mathbf{p}[i]$ to $\mathbf{q}[j]$

$\mathbf{h} \Leftarrow \{0\}^{n\times m}$ ▷ $\mathbf{h}[i][j]$ stores the backwards trace history.

for $j = 0, \cdots, m-1$ **do** ▷ Dynamicprogramming start

  $\mathbf{f}[0][j] \Leftarrow ||\mathbf{p}[0] - \mathbf{q}[j]||_2$

end for

for $i = 1, \cdots, n-1$ **do**

  $u \Leftarrow \mathbf{p}[i] - \mathbf{p}[i-1]$

  for $j = 0 \cdots m-1$ **do**

    $w \Leftarrow \mathbf{p}[i] - \mathbf{q}[j]$

    for $k = 0 \cdots m-1$ **do**

      $v \Leftarrow \mathbf{q}[j] - \mathbf{q}[k]$

      if $\mathbf{f}[i-1][k] + ||w||_2 + ||u-v||_2 < \mathbf{f}[i][j]$ **then**

        $\mathbf{f}[i][j] \Leftarrow \mathbf{f}[i-1][k] + ||w||_2 + ||u-v||_2$

        $\mathbf{h}[i-1][j] \Leftarrow k$

      end if

    end for

  end for

end for

$j \Leftarrow 0$ ▷ Backwards tracing start

for $k = 0, \cdots, m$ **do**

  if $\mathbf{f}[i][k] < \mathbf{f}[i][j]$ **then**

    $j \Leftarrow k$

  end if

end for

for $i = n-2, \cdots, 0$ **do**

  $\hat{\mathbf{p}}[i] \Leftarrow \mathbf{q}[j]$

  $j \Leftarrow \mathbf{h}[i][j]$

end for

Graph updating. Given a pair of time steps i and j, the deformation graph denoted as $G^{\text{d},i,j}$ is initialized, along with G^i and G^j, as mentioned above. During the iterative updating, G^i is first matched to the 3D scatter points $\mathbf{p}^{\text{g},j}$ and 2D mask points to obtain $\mathbf{p}^{\hat{\text{g}},i}$. The loss function is defined as $||\text{warp}(p^{\text{g},i}) - \mathbf{p}^{\hat{\text{g}},i}||$, and the L-BFGS optimizer[36] is used for updates in each iteration. Once the iterative updating is finished, the final correspondences are used to gather radii from 2D images and G^j and to average the radius estimation.

Performance evaluation details

Evaluation metrics for synthetic dataset. All the compared learning-based methods are trained in $\mathcal{D}^{\text{train}}$ and $\mathcal{D}^{\text{p}}$ if viewing poses are required. For evaluation in synthetic dataset, all comparisons are made in previously introduced $\mathcal{D}^{\text{se}}$. The 3D evaluation metrics for $\mathcal{D}^{\text{se}}$ include DICE score, cl-DICE score and Chamfer distance, which are introduced as follows. We denote the flattened vector of a predicted grid (image or volume) and its corresponding ground-truth grid as $\text{pred} \in \{0,1\}^k$, $\text{gt} \in \{0,1\}^k$, where k is the dimension of the vector.

The DICE score is defined as follows:

$$\text{DICE} = \frac{2\sum(\text{pred}_i \cdot \text{gt}_i)}{\sum \text{pred}_i + \sum \text{gt}_i} \tag{17}$$

The skeleton of predicted and ground-truth vessels can be obtained using morphological thinning, denoted as $\text{pred}^{\text{cl}} \in \{0,1\}^k$, $\text{gt}^{\text{cl}} \in \{0,1\}^k$. The cl-DICE score is defined as follows:

$$\text{cl-DICE} = \frac{2 \cdot \frac{\sum(\text{pred}_i \cdot \text{gt}_i^{\text{cl}})}{\sum \text{gt}_i^{\text{cl}}} \cdot \frac{\sum(\text{pred}_i^{\text{cl}} \cdot \text{gt}_i)}{\sum \text{pred}_i^{\text{cl}}}}{\frac{\sum(\text{pred}_i \cdot \text{gt}_i^{\text{cl}})}{\sum \text{gt}_i^{\text{cl}}} + \frac{\sum(\text{pred}_i^{\text{cl}} \cdot \text{gt}_i)}{\sum \text{pred}_i^{\text{cl}}}} \tag{18}$$

The surface point of prediction result and its ground truth can be obtained by marching cube algorithm, denoted as $\text{pred}^{\text{s}} \in \mathbb{R}^{n\times 3}$ and $\text{gt}^{\text{s}} \in \mathbb{R}^{m\times 3}$, where n and m are the numbers of surface points. The distance of any pair of ith predicted points and jth ground-truth points can be obtained by $\|\text{pred}^{\text{s}}[i] - \text{gt}^{\text{s}}[j]\|_2$. The Chamfer distance is defined as:

$$\text{CD} = \frac{1}{n}\sum_{p\in\text{pred}^{\text{s}}} \inf_{q\in\text{gt}^{\text{s}}} \| p - q\|_2 + \frac{1}{m}\sum_{q\in\text{gt}^{\text{s}}} \inf_{p\in\text{pred}^{\text{s}}} \| p - q\|_2 \tag{19}$$

Following the same definition of Chamfer distance on pred^{s} and gt^{s}, with $Q_\alpha(\cdot)$ denotes the α-quantile, the 95% Hausdorff distance is defined as:

$$\begin{aligned}\text{HD95} = \max\Bigg\{&Q_\alpha\left(\left\{\inf_{p\in\text{pred}^{\text{s}}} \| p - q\|_2 : q \in \text{gt}^{\text{s}}\right\}\right),\\ &Q_\alpha\left(\left\{\inf_{q\in\text{gt}^{\text{s}}} \| p - q\|_2 : p \in \text{pred}^{\text{s}}\right\}\right)\Bigg\}.\end{aligned} \tag{20}$$

In the implementation, distance calculation and neighbour query are optimized with KDTree for efficiency.

Evaluation metrics for real-world dataset. For evaluation in real-world scenario, we first introduce the construction of real-world evaluation dataset $\mathcal{D}^{\text{re}}$ and then provide the definition of evaluation metrics. As most of compared methods assume that the key-frame selection or 2D

segmentation has been done by existing tools, we apply the same key-frame selection and 2D segmentation steps to all compared methods if applicable. More specifically, the real-world evaluation set is defined as:

$$\mathcal{D}^{\mathrm{re}} = \{(I^{\mathrm{A},x}, I^{\mathrm{B},x}, I^{\mathrm{A}}, I^{\mathrm{B}}, P^{\mathrm{A}}, P^{\mathrm{B}}), (\mathbf{p}^{\mathrm{A}}, \mathbf{p}^{\mathrm{B}})\}, \tag{21}$$

where $I^{\mathrm{A},x}$, $I^{\mathrm{B},x}$, I^{A}, I^{B}, P^{A} and P^{B} are are the inputs and share the same definitions in previous subsection, and $\mathbf{p}^{\mathrm{A}}, \mathbf{p}^{\mathrm{B}} \in \mathbb{R}^{n_a \times 2}$ are annotated corresponding pixel locations in $I^{\mathrm{A},x}$, $I^{\mathrm{B},x}$, for example $\mathbf{p}_1^{\mathrm{A}}$ and $\mathbf{p}^{\mathrm{B}}[1]$ correspond to the same 3D location. In addition, the predicted surface or centreline points are denoted as $\mathbf{p}^{\mathrm{pred}} \in \mathbb{R}^{n_{\mathrm{pred}} \times 3}$, and its 2D projection are denoted as $\mathbf{p}^{\mathrm{pred,A}}, \mathbf{p}^{\mathrm{pred,B}} \in \mathbb{R}^{n_{\mathrm{pred}} \times 2}$.

The evaluation of correspondence is based on the same correspondence query problem: given a query point, for example $\mathbf{p}_1^{\mathrm{A}}$ in I^{A}, find the corresponding location in I^{B}. There are three aspects considering the quality of predicted correspondence.

The coverage measures the percentages of query point ($\mathbf{p}^{\mathrm{A}}$) that are correctly predicted. This is essential because, when the reconstructed 3D result ($\mathbf{p}^{\mathrm{pred}}$) is incomplete, there might be no predicted query point ($\mathbf{p}^{\mathrm{pred,A}}$) surrounding the ground-truth query point ($\mathbf{p}^{\mathrm{A}}$).

$$\text{Coverage} = \frac{1}{n_a}\sum_{i=1}^{n_a} \mathbf{1}\left(\min_j \| \mathbf{p}_i^{\mathrm{A}} - \mathbf{p}_j^{\mathrm{pred,A}} \|_2 < \epsilon_{\mathrm{v}}\right) \tag{22}$$

The EPE measures the distance between predicted corresponding point ($\mathbf{p}^{\mathrm{pred,B}}$) in another view, and it's ground-truth corresponding point ($\mathbf{p}^{\mathrm{B}}$). The index of predicted corresponding point can be obtained by $\text{index}_i = \arg\min_j \| \mathbf{p}_i^{\mathrm{B}} - \mathbf{p}_j^{\mathrm{pred,B}} \|_2$. Then, the EPE can be calculated as follows:

$$\text{EPE} = \frac{1}{n_a}\sum_{i=1}^{n_a} \| \mathbf{p}_i^{\mathrm{B}} - \mathbf{p}_{\text{index}_i}^{\mathrm{pred,B}} \|_2 \tag{23}$$

The accuracy is defined as the percentage of predicted corresponding point ($\mathbf{p}_{\text{index}_i}^{\mathrm{pred,B}}$) with error value less than ϵ_{v}.

$$\text{Accuracy} = \text{Coverage} \cdot \frac{1}{n_a}\sum_{i=1}^{n_a} \mathbf{1}\left(\| \mathbf{p}_i^{\mathrm{B}} - \mathbf{p}_{\text{index}_i}^{\mathrm{pred,B}} \|_2 < \epsilon_{\mathrm{v}}\right) \tag{24}$$

Estimation of radiation dosage. We calculate the radiation dose using the number of X-ray images, following the method described in previous studies[4,24,25]. For 3D model reconstruction of a single moment, the standard procedure for cerebral tomography, which we use as a reference due to the lack of a specific protocol for C-arm-based coronary tomography, requires 133 projections. However, our AutoCAR system requires only two projections. Therefore, the estimated radiation dosage reduction is 66×.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All data generated and analysed in this study, including the training dataset, synthetic and real-world evaluation dataset, pretrained weights and model predictions of existing method and proposed method results, are publicly available at under a CC BY-NC-ND 4.0 license at GitHub (https://github.com/zhuyinheng/autocar_release) and Zenodo[50] (https://doi.org/10.5281/zenodo.15004536). Please follow the instructions in either public repository for data preparation, reproduction (including training or inference with the provided weights), and for inspecting and comparing the predictions of the proposed method, existing methods and ground truth.

Code availability

The source code is publicly available under a CC BY-NC-ND 4.0 license at GitHub (https://github.com/zhuyinheng/autocar_release) and Zenodo[50] (https://doi.org/10.5281/zenodo.15004536). Please follow the instructions in either public repository for data preparation, reproduction (including training or inference with the provided weights), and for inspecting and comparing the predictions of the proposed method, existing methods and ground truth.

References

1. Mézquita, A. J. V. et al. Clinical quantitative coronary artery stenosis and coronary atherosclerosis imaging: a consensus statement from the quantitative cardiovascular imaging study group. *Nat. Rev. Cardiol.* **20**, 696–714 (2023).
2. Mahmoud, K. D. & Zijlstra, F. Thrombus aspiration in acute myocardial infarction. *Nat. Rev. Cardiol.* **13**, 418–428 (2016).
3. Çimen, S., Gooya, A., Grass, M. & Frangi, A. F. Reconstruction of coronary arteries from X-ray angiography: a review. *Med. Image Anal.* **32**, 46–68 (2016).
4. Zhao, H. et al. Self-supervised learning enables 3D digital subtraction angiography reconstruction from ultra-sparse 2D projection views: a multicenter study. *Cell Rep. Med.* **3**, 100775 (2022).
5. Feldkamp, L. A., Davis, L. C. & Kress, J. W. Practical cone-beam algorithm. *J. Opt. Soc. Am. A* **1**, 612 (1984).
6. Neubauer, A. M. et al. Clinical feasibility of a fully automated 3D reconstruction of rotational coronary X-ray angiograms. *Circ. Cardiovasc. Interv.* **3**, 71–79 (2010).
7. Liu, J. et al. 5D respiratory motion model based image reconstruction algorithm for 4D cone-beam computed tomography. *Inverse Probl.* **31**, 115007 (2015).
8. Blondel, C., Malandain, G., Vaillant, R. & Ayache, N. Reconstruction of coronary arteries from a single rotational X-ray projection sequence. *IEEE Trans. Med. Imaging* **25**, 653–663 (2006).
9. Unberath, M., Taubmann, O., Hell, M., Achenbach, S. & Maier, A. Symmetry, outliers, and geodesics in coronary artery centerline reconstruction from rotational angiography. *Med. Phys.* **44**, 5672–5685 (2017).
10. Banerjee, A. et al. Point-cloud method for automated 3D coronary tree reconstruction from multiple non-simultaneous angiographic projections. *IEEE Trans. Med. Imaging* **39**, 1278–1290 (2020).
11. Cong, W. et al. Quantitative analysis of deformable model-based 3-D reconstruction of coronary artery from multiple angiograms. *IEEE Trans. Biomed. Eng.* **62**, 2079–2090 (2015).
12. Yang, J. et al. External force back-projective composition and globally deformable optimization for 3-D coronary artery reconstruction. *Phys. Med. Biol.* **59**, 975–1003 (2014).
13. Liao, R., Luc, D., Sun, Y. & Kirchberg, K. 3-D reconstruction of the coronary artery tree from multiple views of a rotational X-ray angiography. *Int. J. Cardiovasc. Imaging* **26**, 733–749 (2010).
14. Zheng, S., Meiying, T. & Jian, S. Sequential reconstruction of vessel skeletons from X-ray coronary angiographic sequences. *Comput. Med. Imaging Graph.* **34**, 333–345 (2010).
15. Yang, J., Wang, Y., Liu, Y., Tang, S. & Chen, W. Novel approach for 3-D reconstruction of coronary arteries from two uncalibrated angiographic images. *IEEE Trans. Image Process.* **18**, 1563–1572 (2009).
16. Cong, W., Yang, J., Liu, Y. & Wang, Y. Energy back-projective composition for 3-D coronary artery reconstruction. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 5151–5154 (IEEE, 2013).
17. Galassi, F. et al. 3D reconstruction of coronary arteries from 2D angiographic projections using non-uniform rational basis splines (NURBS) for accurate modelling of coronary stenoses. *PLoS ONE* **13**, e0190650 (2018).

18. Hwang, M. et al. A simple method for automatic 3D reconstruction of coronary arteries from X-ray angiography. *Front. Physiol.* **12**, 724216 (2021).
19. Bransby, K. M. et al. 3D coronary vessel reconstruction from bi-plane angiography using graph convolutional networks. In *IEEE International Symposium on Biomedical Imaging* 1–5 (IEEE, 2023).
20. Wang, G., Ye, J. C. & De Man, B. Deep learning for tomographic image reconstruction. *Nat. Mach. Intell.* **2**, 737–748 (2020).
21. Iyer, K., Nallamothu, B. K., Figueroa, C. A. & Nadakuditi, R. R. A multi-stage neural network approach for coronary 3D reconstruction from uncalibrated X-ray angiography images. *Sci. Rep.* **13**, 17603 (2023).
22. Liang, D. & Chen, S. Weakly-supervised 3D coronary artery reconstruction from two-view angiographic images. Preprint at *Research Square* https://doi.org/10.21203/rs.3.rs-3703340/v1 (2023).
23. Gao, C. et al. Synthetic data accelerates the development of generalizable learning-based algorithms for X-ray image analysis. *Nat. Mach. Intell.* **5**, 294–308 (2023).
24. Ying, X. et al. X2CT-GAN: reconstructing CT from biplanar X-rays with generative adversarial networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 10619–10628 (IEEE, 2019).
25. Kasten, Y., Doktofsky, D. & Kovler, I. End-to-end convolutional neural network for 3D reconstruction of knee bones from bi-planar X-ray images. In *International Workshop on Machine Learning for Medical Image Reconstruction* 123–133 (Springer, 2020).
26. Kini, A. & Sharma, S. K. (eds) *Practical Manual of Interventional Cardiology* (Springer, 2021).
27. Graham, B., Engelcke, M. & Van Der Maaten, L. 3D semantic segmentation with submanifold sparse convolutional networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 9224–9232 (IEEE, 2018).
28. Shen, L., Zhao, W. & Xing, L. Patient-specific reconstruction of volumetric computed tomography images from a single projection view via deep learning. *Nat. Biomed. Eng.* **3**, 880–888 (2019).
29. Li, M., Yang, H. & Kudo, H. An accurate iterative reconstruction algorithm for sparse objects: application to 3D blood vessel reconstruction from a limited number of projections. *Phys. Med. Biol.* **47**, 2599 (2002).
30. Hansis, E., Schafer, D., Dossel, O. & Grass, M. Evaluation of iterative sparse object reconstruction from few projections for 3-D rotational coronary angiography. *IEEE Trans. Med. Imaging* **27**, 1548–1555 (2008).
31. Jandt, U., Schäfer, D., Grass, M. & Rasche, V. Automatic generation of 3D coronary artery centerlines using rotational X-ray angiography. *Med. Image Anal.* **13**, 846–858 (2009).
32. Unberath, M. et al. Respiratory motion compensation in rotational angiography: graphical model-based optimization of auto-focus measures. In *IEEE International Symposium on Biomedical Imaging* 227–230 (IEEE, 2017).
33. Antiga, L., Ene-Iordache, B. & Remuzzi, A. Computational geometry for patient-specific reconstruction and meshing of blood vessels from MR and CT angiography. *IEEE Trans. Med. Imaging* **22**, 674–684 (2003).
34. Zheng, J.-Q., Zhou, X.-Y., Riga, C. & Yang, G.-Z. Towards 3D path planning from a single 2D fluoroscopic image for robot assisted fenestrated endovascular aortic repair. In *International Conference on Robotics and Automation* 8747–8753 (IEEE, 2019).
35. Sumner, R. W., Schmid, J. & Pauly, M. Embedded deformation for shape manipulation. *ACM Trans. Graph.* **26**, 80 (2007).
36. Habert, S., Habert, S., Dahdah, N. & Cheriet, F. A novel method for an automatic 3D reconstruction of coronary arteries from angiographic images. In *International Conference on Information Science, Signal Processing and their Applications* 484–489 (IEEE, 2012).
37. Shit, S. et al. clDICE—a novel topology-preserving loss function for tubular structure segmentation. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 16560–16569 (IEEE, 2021).
38. Royer-Rivard, R., Girard, F., Dahdah, N. & Cheriet, F. End-to-end deep learning model for cardiac cycle synchronization from multi-view angiographic sequences. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 1190–1193 (IEEE, 2020).
39. Vlontzos, A. & Mikolajczyk, K. Deep segmentation and registration in X-ray angiography video. In *British Machine Vision Conference* 267–278 (BMVA, 2018).
40. Schonberger, J. L. & Frahm, J.-M. Structure-from-motion revisited. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 4104–4113 (IEEE, 2016).
41. Yu, T. et al. DoubleFusion: real-time capture of human performances with inner body shapes from a single depth sensor. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**, 2523–2539 (2019).
42. Rohkohl, C., Lauritsch, G., Keil, A. & Hornegger, J. CAVAREV—an open platform for evaluating 3D and 4D cardiac vasculature reconstruction. *Phys. Med. Biol.* **55**, 2905 (2010).
43. Zeng, A. et al. ImageCAS: a large-scale dataset and benchmark for coronary artery segmentation based on computed tomography angiography images. *Comput. Med. Imaging Graph.* **109**, 102287 (2023).
44. Antiga, L., Ene-Iordache, B., Remuzzi, G. & Remuzzi, A. Automatic generation of glomerular capillary topological organization. *Microvasc. Res.* **62**, 346–354 (2001).
45. Hartley, R. & Zisserman, A. *Multiple View Geometry in Computer Vision* (Cambridge Univ. Press, 2003).
46. Ravi, N. et al. Accelerating 3D deep learning with PyTorch3D. Preprint at https://arxiv.org/abs/2007.08501 (2020).
47. Riba, E., Mishkin, D., Ponsa, D., Rublee, E. & Bradski, G. Kornia: an open source differentiable computer vision library for PyTorch. In *Proc. IEEE/CVF Winter Conference on Applications of Computer Vision* 3674–3683 (IEEE, 2020).
48. Newell, A., Yang, K. & Deng, J. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision* 483–499 (2016).
49. Gwak, J., Choy, C. & Savarese, S. Generative sparse detection networks for 3D single-shot object detection. In *European Conference on Computer Vision* 297–313 (2020).
50. Zhu, Y. et al. Sparse and transferable 3D dynamic vascular reconstruction for instantaneous diagnosis. *Zenodo* https://doi.org/10.5281/zenodo.15004536 (2025).

Acknowledgements

We thank Y. Yang and Z. Guo for XA pre-processing advice; K. Hu for advices in visualization; Y. Zheng for clinical advice; and H. Xu, Z. Zhang and L. Yang for discussion. This work was supported by National Natural Science Foundation of China (grant number 32371470,82341019, S.M.), National Key Research and Development Program of China (2024YFA0919800, S.M.), Department of Science and Technology of Guangdong Province (grant number 2023B0909020003, S.M.), and Cross-disciplinary Research and Innovation Fund of Tsinghua SIGS (grant number JC2022007, S.M.).

Author contributions

S.M. and Y.Z. conceived of the project. Y.Z. designed the methodology of the reconstruction algorithms with suggestions from F.L. Y.Z. conducted the main experiments and developed the software. Y.Z. and C.D. performed data curation. Y.Z., H.L. and Y.W. conducted the validation experiments. Y.Z., S.M. and Y.W. prepared the figures and

visualization, and wrote the paper with input from all authors. The whole project was supervised by S.M.

Competing interests

Y.Z., S.M. and F.L. have applied for a patent related to the work reported in this paper. The other authors declare no competing interests.

Additional information

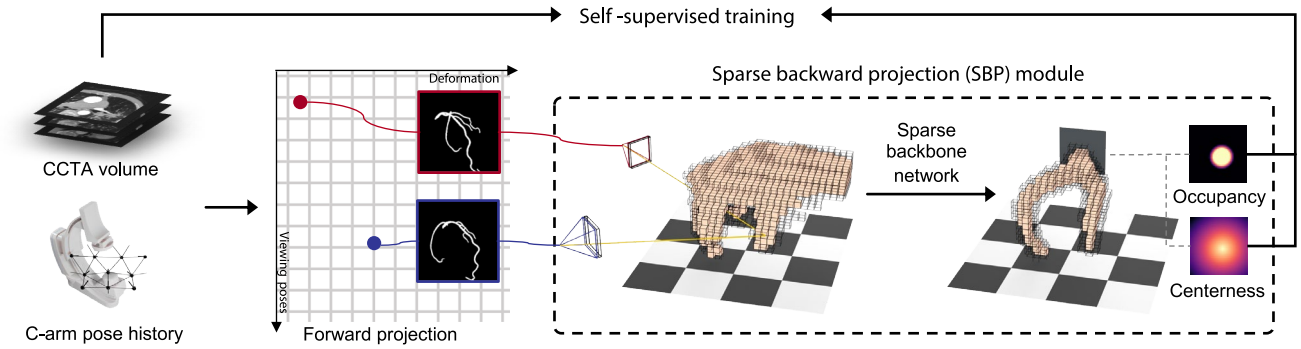
Extended data is available for this paper at https://doi.org/10.1038/s42256-025-01025-7.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-025-01025-7.

Correspondence and requests for materials should be addressed to Fangzhou Liao or Shaohua Ma.

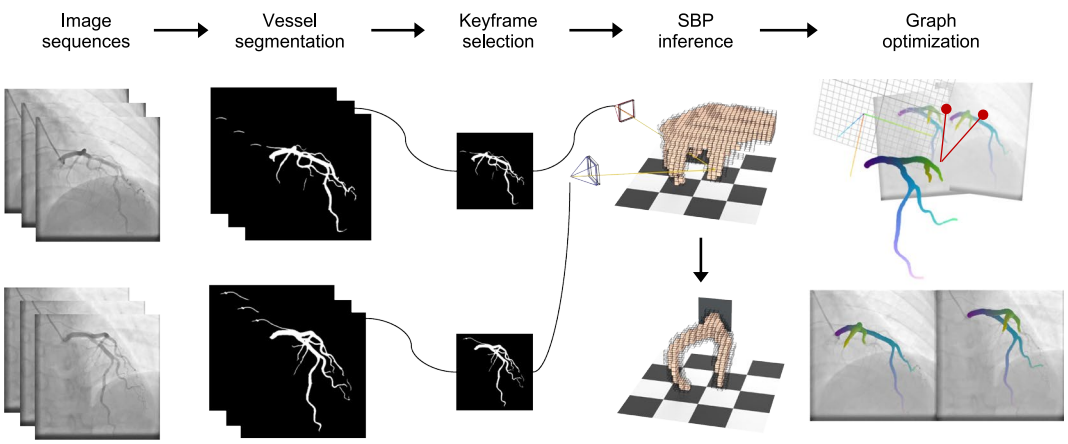
Peer review information *Nature Machine Intelligence* thanks Haoran Dou, Jan Egger, Alejandro Frangi and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

https://doi.org/10.1038/s42256-025-01025-7



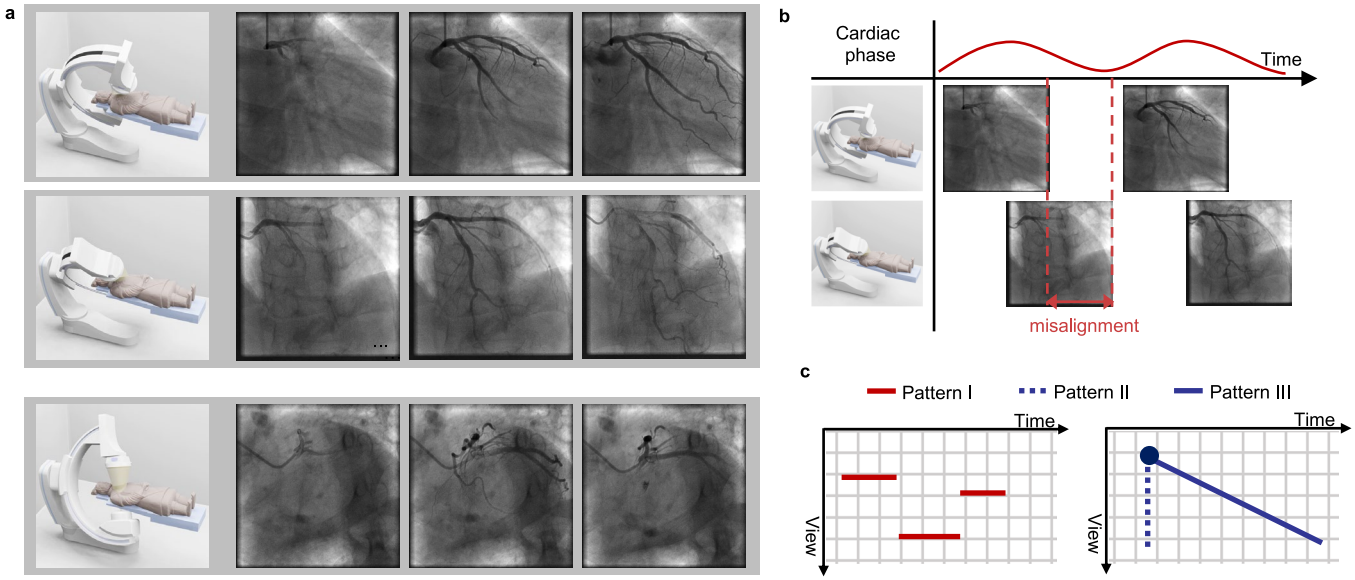
Extended Data Fig. 1 | The training pipeline of AutoCAR. Synthetic XA images are generated by forward-projecting the 3D coronary surface mesh from the CCTA dataset with given imaging parameters. The projected results are naturally vascular mask images and processed by the SBP module. Specifically, two mask images are backward-projected into 3D space, fused into one 3D volume, and processed by the sparse backbone network to output per-voxel vascular occupancy and centerness. During the inference process, only the part in the dashed box (that is, the SBP module) needs to be performed.

https://doi.org/10.1038/s42256-025-01025-7



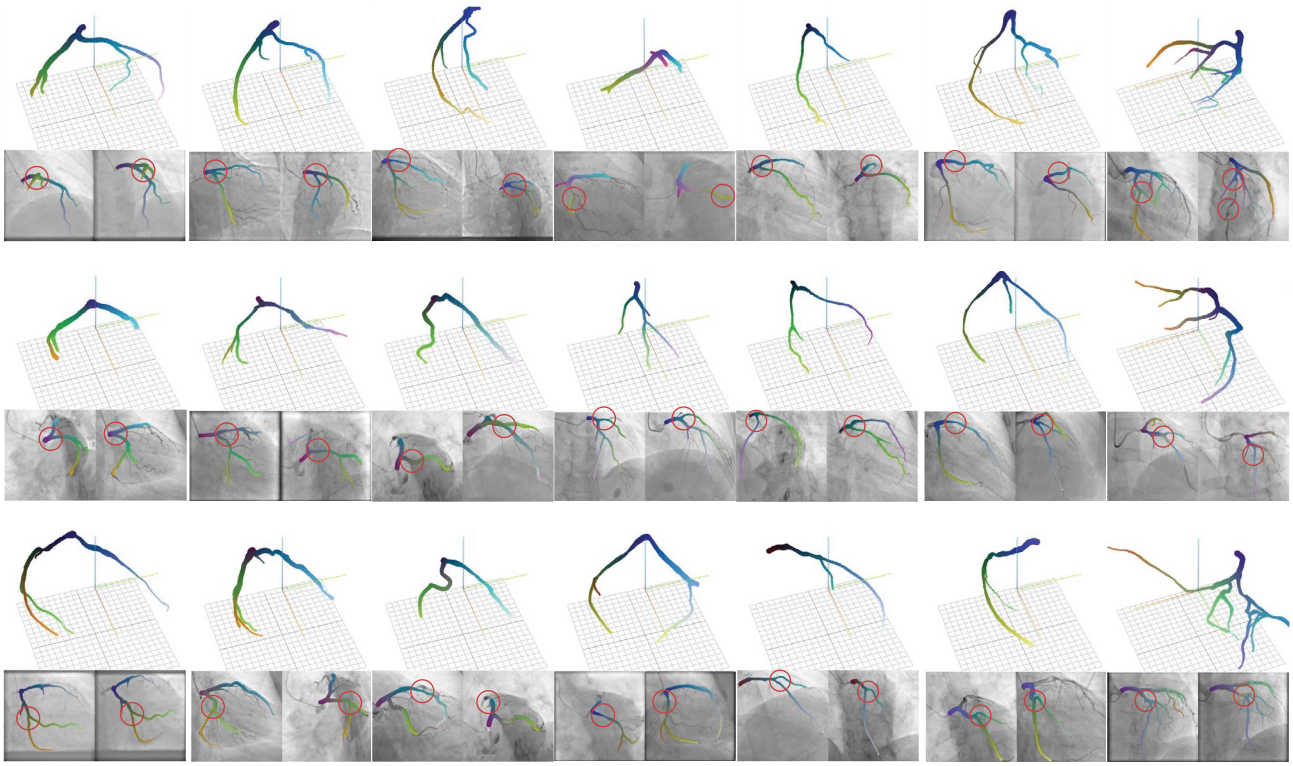
Extended Data Fig. 2 | The inference pipeline of AutoCAR. AutoCAR directly receives two image sequences captured from different views as input, and gives the 3D reconstruction result in a fully-automated manner.

https://doi.org/10.1038/s42256-025-01025-7



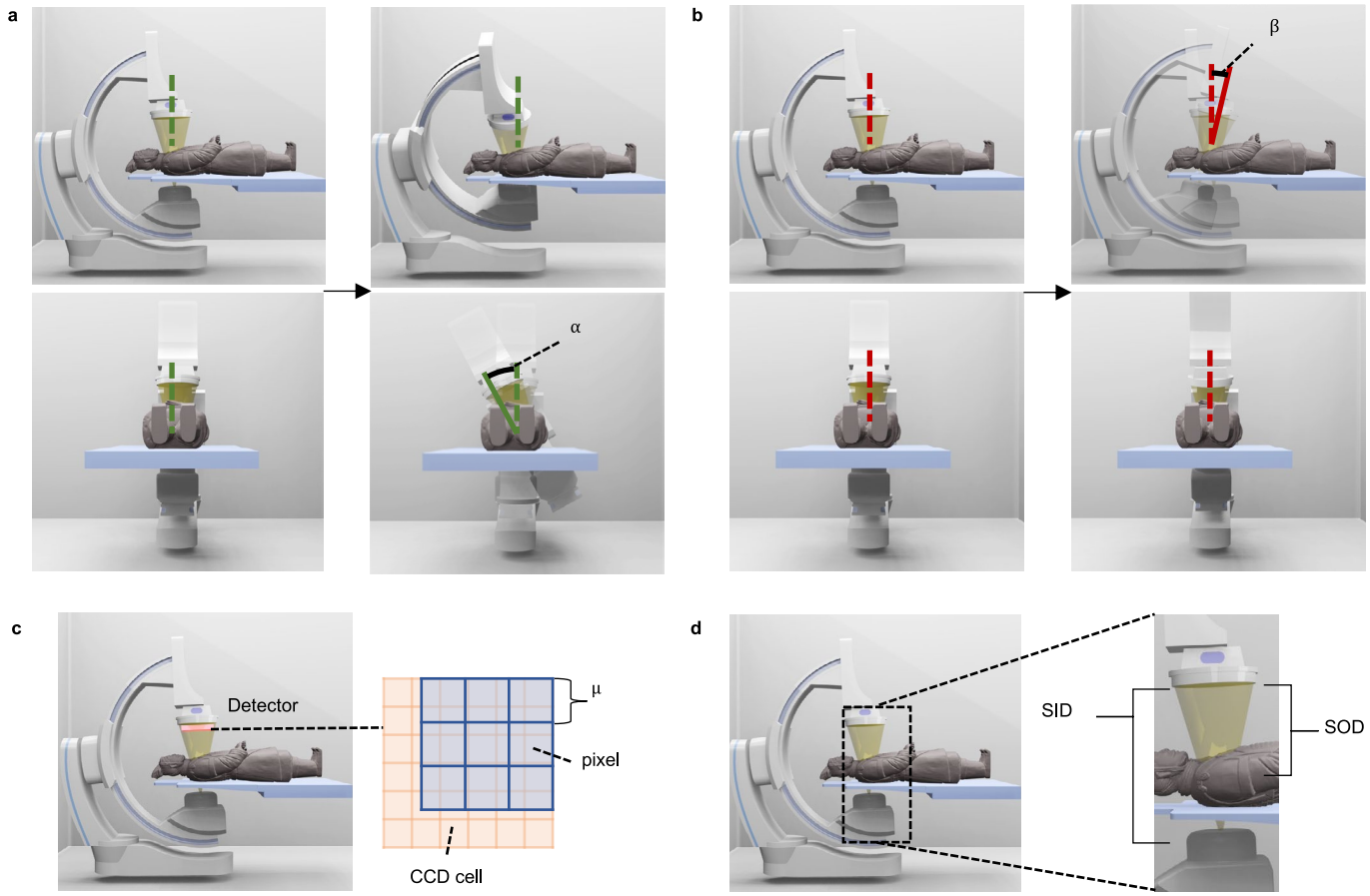
Extended Data Fig. 3 | Weakly synchronization property and infeasibility of 3D ground truths. a, Scanning protocol and exemplar input. The rows correspond to different viewing poses, while the first column indicates the C-Arm rotation state and the exemplar frames in X-ray angiography are shown starting from the second column. Due to the deformation of cardiovasculature structures, instead of using a continuous circular scanning approach common in head and neck angiography, the protocol recommends capturing a video from a fixed viewpoint before transitioning to the next, yielding multiple multi-view image sequences. **b**, Weak synchronization issues. Even with ECG for synchronization, and with patients holding their breath, which is often challenging for emergencies, with elderly/young patients or heart rate variability, temporal misalignment persist. **c**, Challenges in paired multi-view 2D X-ray angiography and 3D ground truth. Each dot in view time space represents an image at a unique pose and time, and a sequence forms a scanning pattern (Pattern I, red segments). For 3D reconstruction of blue-dot image, simultaneous multi-view images (Pattern II) are essential. However, capturing images within 1/6 of a cardiac cycle would require the C-arm to rotate 1080 deg/s and record at 900 fps (Pattern III), far exceeding current systems like Siemens Axiom Artis at 60 fps.

https://doi.org/10.1038/s42256-025-01025-7



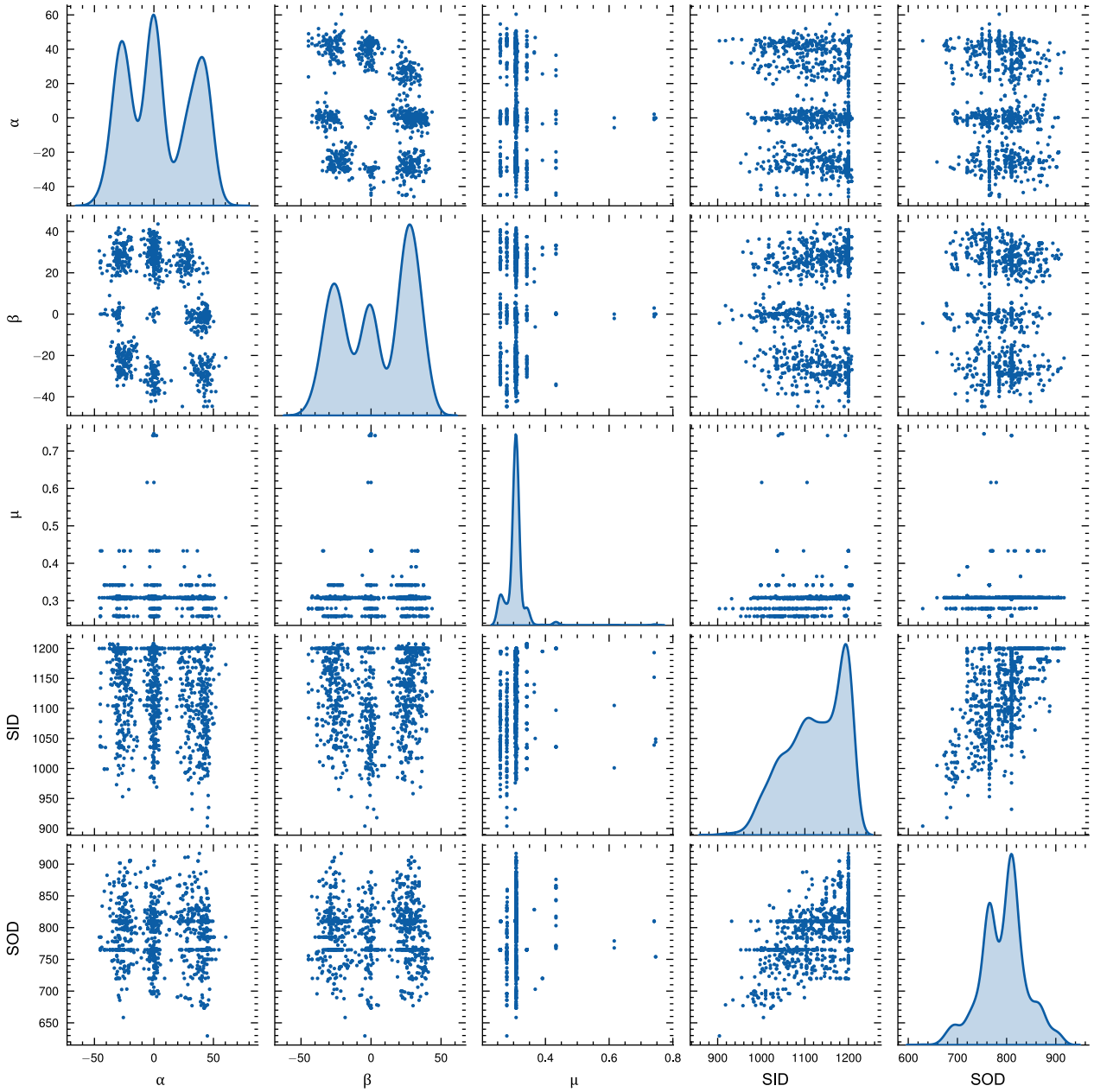
Extended Data Fig. 4 | AutoCAR reconstruction of 3D Models and 2D Correspondences Across 21 Cases. For each case, the top section displays the 3D visualization, while the left and right images provide views from two distinct perspectives. The areas highlighted by the red circle denotes suspicious regions, where clinicians may wish to investigate the 3D shapes or their corresponding 2D regions from a different view angle. The colour coding is used to indicate correspondence among 3D and multi-views for each individual cases.

https://doi.org/10.1038/s42256-025-01025-7



Extended Data Fig. 5 | Illustration of imaging parameters in DICOM. a, Primary angle α is defined as the rotation angle with respect to craniocaudal axis. Positive direction is defined at the patient left hand side. **b**, Secondary angle β is defined as the rotation angle with respect to secondary axis, which is in the patient plane (horizontal) and is perpendicular to the craniocaudal axis at the isocentre. Cranial direction is regarded as positive direction. **c**, Pixel spacing μ is defined as the physical length of each pixel. **d**, Source intensifier distance (SID) is defined as the distance between intensifier (detector) and X-ray source. Source object distance (SOD) is defined as the distance between isocentre and X-ray source. The above definitions are consistent with DICOM standard (C.8.7.5.1.2, Section 10.7.1.1).

https://doi.org/10.1038/s42256-025-01025-7



Extended Data Fig. 6 | Pairwise scatterplot of imaging parameter distributions. α, β, μ denote primary angle, secondary angle and pixel spacing, respectively. All statistics are summarized from the imaging parameters of 1215 real-world XA cases.

https://doi.org/10.1038/s42256-025-01025-7

Extended Data Table 1 | Configurations of learning-based methods

Config ID	Method	BP	SBP	Pose	Proj.	Rec.	Adv.
L1	Coronary or Lung [22, 28]	×	×	×	×	✓	×
L2	Knee bones[25]	✓	×	×	×	✓	×
L3	Lung[24]	✓	×	×	✓	✓	✓
L4	Head and neck vessels[4]	✓	×	×	✓	×	×
L5	Ablation study	✓	×	×	×	✓	✓
L6	Ablation study	✓	×	×	✓	✓	×
L7	Ablation study	✓	✓	×	×	✓	×
L8	AutoCAR	✓	✓	✓	×	✓	×

BP, SBP, Pose are abbreviations of backward projection, sparse backward projection, pose domain adaptation. Proj., Rec. and Adv. are abbreviations of projection, reconstruction and adversarial loss. ×, ✓ denotes "without" and "with" respectively.

https://doi.org/10.1038/s42256-025-01025-7

Extended Data Table 2 | Expert annotation of stenosis

Case id	Stenosis
1	Severe stenosis in the proximal segment of the LAD, mild stenosis in the mid segment of the LAD
2	Moderate stenosis in the mid segment of the LAD
3	Moderate to severe stenosis at the origin of the LAD, severe stenosis in the mid segment of the LCX
4	Severe stenosis in the proximal segment of the LAD
5	Severe stenosis in the mid segment of the LAD, severe stenosis at the origin of the D1, severe stenosis at the origin of the LCX, moderate stenosis in the mid segment of the LCX
6	Moderate stenosis in the mid segment of the LAD
7	Severe stenosis at the opening of the LAD, severe stenosis in the mid segment of the LCX, severe stenosis in the intermediate branch
8	Severe stenosis in the proximal to mid segment of the LAD
9	Severe stenosis in the proximal to mid segment of the LAD
10	Moderate to severe stenosis in the proximal segment of the LAD, severe stenosis in the mid to distal segment of the LCX, severe stenosis in the OM2
11	Mild stenosis in the proximal segment of the LAD, severe stenosis in the mid segment
12	Mild stenosis in the proximal segment of the LAD, severe stenosis in the proximal to mid segment of the LCX
13	Mild stenosis in the proximal segment of the LAD, severe stenosis in the mid segment, moderate stenosis in the proximal to mid segment of the LCX
14	Mild stenosis in the proximal segment of the LAD, severe stenosis in the mid segment, moderate stenosis in the proximal to mid segment of the LCX
15	Severe stenosis in the mid segment of the LAD, severe stenosis at the origin of the D1, severe stenosis at the origin of the LCX
16	Severe stenosis in the mid segment of the LAD, light to moderate stenosis at the origin of the LCX, severe stenosis in the mid segment of the LCX
17	Severe stenosis in the LAD, moderate stenosis in the LCX
18	Moderate stenosis in the mid segment of the LAD, mild stenosis in the mid segment of the LCX
19	Multiple severe stenoses in the proximal to mid segment of the LAD, mild stenosis at the origin of the D1, light to moderate stenosis in the mid segment of the LCX
20	Severe stenosis in the proximal segment of the LAD, severe stenosis in the proximal segment of the D1, moderate stenosis in the mid segment of the LCX
21	Severe stenosis in the proximal segment of the LAD, severe stenosis at the origin of the diagonal branch; mild stenosis at the origin of the LCX, moderate stenosis in the mid segment of LCX

"LAD", "LCX", "D1", "OM2" refers to Left Anterior Descending artery, Left Circumflex artery, first Diagonal and second Obtuse Marginal branch respectively.

nature portfolio

Corresponding author(s): Shaohua MA, Fangzhou LIAO

Last updated by author(s): Mar 12, 2025

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☐	☒	A description of all covariates tested
☐	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☐	☒	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection: no software was used for data collection.

Data analysis	Neural network–related libraries include (1) Minkowski Engine v0.5.4 (https://github.com/NVIDIA/MinkowskiEngine) for sparse operators, (2) PyTorch v1.10 (https://pytorch.org) for dense operators, (3) PyTorch3D v0.7 for SO(3)-related utilities and rendering during training, (4) Hourglass Network v-commit-ceedc14 (https://github.com/princeton-vl/pytorch_stacked_hourglass) for the 2D hourglass network architecture, (5) Kornia Library v0.7.0 (https://github.com/kornia/kornia) for thin plate spline deformation, and (6) PyTorch-LBFGS v1.10.0 (https://pytorch.org, https://github.com/hjmshi/PyTorch-LBFGS) for the LBFGS optimizer in graph optimization. Other main libraries and tools include (1) dijkstra3d v1.13 (https://github.com/seung-lab/dijkstra3d) for volumetric Dijkstra's shortest path, (2) networkx v3.4.0 (https://github.com/networkx/networkx) for graph representation and minimal spanning tree implementation, (3) scipy v1.11 for operators such as upsampling, Gaussian filtering, and marching cubes, (4) COLMAP v3.8 (https://github.com/colmap/colmap) for implementing Patch Match Stereo (an existing method), (5) Blender (https://github.com/blender/blender) and matplotlib (https://github.com/matplotlib/matplotlib) for data visualization. For a complete list of libraries used, please refer to the project's GitHub repository (custom codebase, see below), where versions and dependencies are automatically resolved. Custom codebase: The source code is publicly available under the CC BY-NC-ND 4.0 License at GitHub (https://github.com/zhuyinheng/autocar_release) and Zenodo (https://doi.org/10.5281/zenodo.15004536). Please follow the instructions in for data preparation, reproduction (including training or inference with the provided weights), and for inspecting and comparing the predictions of the proposed method, existing methods, and ground truth.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

All data generated and analyzed in this study, including the training dataset, synthetic and real-world evaluation dataset, pre-trained weights and model predictions of existing method and proposed method results, are publicly available at under the CC BY-NC-ND 4.0 License at GitHub (https://github.com/zhuyinheng/autocar_release) and Zenodo (https://doi.org/10.5281/zenodo.15004536).

Research involving human participants, their data, or biological material

Policy information about studies with human participants or human data. See also policy information about sex, gender (identity/presentation), and sexual orientation and race, ethnicity and racism.

Reporting on sex and gender	N/A
Reporting on race, ethnicity, or other socially relevant groupings	N/A
Population characteristics	N/A
Recruitment	N/A
Ethics oversight	N/A

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	The evaluation dataset sample size was determined based on previous studies in the field.

Sample size	In [1], 45 patients were included, yielding 107 3D structures. Note that [1] focused on single temporal frame 3D reconstruction, meaning each patient has only one "complete 3D structure," while branches (e.g., LCX, LAD) were counted separately, which explains why the total is 107 instead of 45. In our study, 21 patients were included with 147 3D structures. Our approach targets multiple temporal frame 3D reconstruction (7 frames per patient), and the number of 3D structures is calculated based on arteries—branches such as LCX and LAD are counted as a single 3D structure—so the 147 structures result from 21 patients multiplied by 7 frames. [1] A. Banerjee, F. Galassi, E. Zacur, G. L. De Maria, R. P. Choudhury, V. Grau, Point-Cloud Method for Automated 3D Coronary Tree Reconstruction From Multiple Non-Simultaneous Angiographic Projections. IEEE Trans. Med. Imaging. 39, 1278–1290 (2020).
Data exclusions	No data was excluded from analysis.
Replication	The experiments and findings are replicatable using the source code and data provided at GitHub (https://github.com/zhuyinheng/autocar release) and Zenodo (https://doi.org/10.5281/zenodo.15004536).
Randomization	The training set, evaluation set of D^{CT} are randomly split (without overlap) as described in method section.
Blinding	In our study, blinding was not relevant because the data collection occurred independently of the study design. Specifically, the multi-view images used were X-ray recordings captured during surgical procedures, and our research team did not intervene in or influence the surgeons' actions. Since the images were acquired as part of routine practice prior to our analysis, there was no group allocation during data collection that could introduce bias. Therefore, the need for investigator blinding was inherently negated.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern
☒	☐ Plants

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Plants

Seed stocks	N/A
Novel plant genotypes	N/A
Authentication	N/A