Deep Learning-Based Pulmonary Artery Surface Mesh Generation

Nina Krüger[1,2,3](✉), Jan Brüning[1,2], Leonid Goubergrits[1,2], Matthias Ivantsits[1,2], Lars Walczak[1,2,3], Volkmar Falk[2,4,5], Henryk Dreger[1,2,4], Titus Kühne[1,2,4], and Anja Hennemuth[1,2,3,4,6]

[1] Deutsches Herzzentrum der Charité (DHZC), Institute of Computer-assisted Cardiovascular Medicine, Augustenburger Platz 1, 13353 Berlin, Germany
nina.krueger@dhzc-charite.de

[2] Charité - Universitätsmedizin Berlin, corporate member of Freie Universität Berlin and Humboldt-Universität zu Berlin, Charitéplatz 1, 10117 Berlin, Germany

[3] Fraunhofer MEVIS, Bremen, Germany

[4] DZHK (German Centre for Cardiovascular Research), Partner Site Berlin, Berlin, Germany

[5] Deutsches Herzzentrum der Charité (DHZC), Department of Cardiothoracic and Vascular Surgery, Berlin, Germany

[6] University Medical Center Hamburg-Eppendorf, Department of Diagnostic and Interventional Radiology and Nuclear Medicine, Hamburg, Germany

Abstract. Properties of the pulmonary artery play an essential role in the diagnosis and treatment planning of diseases such as pulmonary hypertension. Patient-specific simulation of hemodynamics can support the planning of interventions. However, the variable complex branching structure of the pulmonary artery poses a challenge for image-based generation of suitable geometries. State-of-the-art segmentation-based approaches require an interactive 3D surface reconstruction to prepare the simulation geometry. We propose a deep learning approach to generate a 3D surface mesh of the pulmonary artery from CT images suitable for simulation. The proposed method is based on the Voxel2Mesh algorithm and includes a voxel encoder and decoder as well as a mesh decoder to deform a prototype mesh. An additional centerline coverage loss facilitates the reconstruction of the branching structure. Furthermore, vertex classification allows for the definition of in- and outlets. Our model was trained with 48 human cases and tested on 10 human cases annotated by two observers. The differences in the anatomical parameters inferred from the automatic surface generation correspond to the differences between the observers' annotations. The suitability of the generated mesh geometries for numerical flow simulations is demonstrated.

Keywords: Graph Neural Network · Voxel2Mesh · Surface Reconstruction · Mesh Generation · Pulmonary Artery

O. Camara et al. (Eds.): STACOM 2023, LNCS 14507, pp. 140–151, 2024.
https://doi.org/10.1007/978-3-031-52448-6_14

1 Introduction

The pulmonary artery (PA) originates from the right ventricle and delivers deoxygenated blood towards the lungs, where it is oxygenated before reaching the left heart and thus the systemic circulation. The blood pressure within the PA is an important biomarker for assessing the severity of heart failure. Modeling and simulation are important tools for investigation of the complex hemodynamics [5,16]. Numerical simulation of PA hemodynamics requires well-defined and high-quality surface meshes enclosing the PA vessel lumen [11]. Pure voxel-based segmentation of the underlying 3D image data [4,10,13,18] is not sufficient, as these suffer from step artifacts depending on the voxel size [3]. When transformed into a surface representation, the resulting meshes often show defects such as bad-quality triangles [2,15]. Using image-based anatomical information for flow simulation typically requires adjustments such as correcting segmentation masks, expanding inflow and outflow regions, or surface smoothing [2]. This can result in time- and resource-intensive interaction.

Data-driven models show promise to unify this entire processing pipeline, allowing to reduce or fully omit manual interaction, but also unnecessary steps, which in turn might reduce the propagation of errors. Kong et al. [8] proposed such a unified approach for mesh generation of the whole heart. However, the level of detail of the mesh representation is limited and considered insufficient for complex vascular structures.

In this study, we demonstrate an approach for the automatic generation of pulmonary artery surface meshes, which can be used for numerical assessment of intra-arterial hemodynamics. This requires the main (MPA), left (LPA) and right (RPA) pulmonary artery up to two degrees of branching vessel. The inlet as well as all outlets (for simplicity referred to as openings) are truncated for subsequent specification of boundary conditions in the numerical setup. We combine different techniques for accurate segmentation, mesh generation, and opening, compare relevant anatomical parameters of automatically and manually obtained surface meshes, and lastly, apply computational fluid dynamics to test the suitability of our surface geometries for hemodynamic assessment.

2 Methods

We propose a deep learning approach based on the Voxel2Mesh algorithm [17] for automatic pulmonary artery mesh generation from contrast-enhanced CT images of the heart. Voxel2Mesh is a neural network architecture, that directly produces 3-dimensional triangular surface meshes from volumetric image data. The Voxel2Mesh algorithm combines information from a simultaneously trained voxel-wise segmentation with a mesh decoder, that deforms a prototype mesh to approximate the target mesh (see Fig. 1). Consequently, the prototype mesh and the target mesh need to have the same topology. As the number of branches of the pulmonary artery differs this cannot be achieved with open-ended surface meshes. Hence, we train the Voxel2Mesh algorithm with closed meshes, using

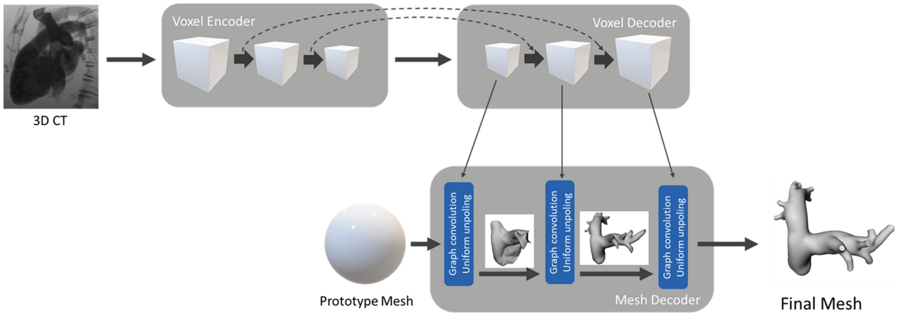

Fig. 1. An overview of the proposed architecture, including a voxel-encoder, a voxel-decoder, and a mesh-decoder: Using a 3D CT image as input, the model successively deforms a prototype mesh at increasing resolution to approximate the target mesh. For this, the mesh-decoder samples features from the latent voxel-space.

a sphere as the prototype. Additionally, our algorithm contains a vertex classification to differentiate between the vessel wall and opening, allowing for the removal of respective faces.

2.1 Data

We used retrospective CT image data of 58 aortic stenosis patients, weighing 76.8 ± 18.2 kg, with an age of 81 ± 7.6 years and a female percentage of 60% for the construction of human pulmonary artery surface meshes. CT data sets of the entire heart were acquired as part of transcatheter aortic valve implantation planning at our clinical center, with wide area-detector volume CT scanners: Aquilion One Vision (Canon Medical Systems, Tochigi, Japan) or Revolution CT (GE Healthcare, Chicago, IL, USA) with 100 kV tube voltage, (0.390–0.648 mm) × (0.390–0.648 mm) in-plane resolution and (0.5–1.0 mm) slice thickness. The institutional review board (Ethikkommission Charité - Universitätsmedizin Berlin, approval number EA2/004/21) approved the use of retrospective data for this study and waived individual informed consent. Figure 2 shows a sample of the reconstruction pipeline from voxel-wise CT segmentation to open-ended surface mesh. We used 40 cases for training the neural network and 8 for validation. A thorough evaluation of anatomical parameters and simulation properties was done on the remaining 10 cases. Two observers generated meshes for each of these 10 cases and we compare deviations from our automatically generated meshes to inter-observer variability. We resampled all CT images to an isotropic resolution of 1.7 mm for neural network training, due to memory constraints. We close all mesh openings, ensuring that the normals are consistently pointing outwards.

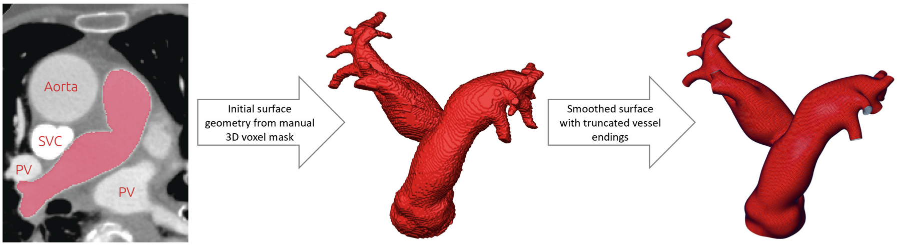
Aorta
SVC
PV
PV
Initial surface geometry from manual 3D voxel mask
Smoothed surface with truncated vessel endings

Fig. 2. Illustration of the pipeline for manual reconstruction of the PA surface mesh from CT image data: The region of interest is segmented manually within the image data and a rough initial surface geometry is reconstructed from the 3D voxel mask. This surface is subsequently smoothed and all vessel endings are truncated. SVC = superior vena cava; PV = pulmonary vein.

2.2 Voxel2Mesh-Based Surface Mesh Generation

Voxel2Mesh uses several loss functions and regularization terms [17]. The cross-entropy loss

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=0}^{1} log \frac{exp(x_{n,c})}{\sum_{i=0}^{1} exp(x_{n,i})} y_{n,c} \tag{1}$$

for training the voxel encoder, where x is the prediction, y is the ground-truth, c is the class (0 for background, 1 for the pulmonary artery) and N is the number of samples within the batch. For training the mesh decoder, the Chamfer distance

$$\mathcal{L}_{CD} = \sum_{v_p \in S_p} \min_{v_g \in S_g} ||\vec{x}_p - \vec{x}_g|| + \sum_{v_g \in S_g} \min_{v_p \in S_p} ||\vec{x}_p - \vec{x}_g|| \tag{2}$$

is applied, where S_p and S_g denote the predicted and ground-truth surfaces, with v_p and v_g being the respective vertices sampled from those surfaces, and $\vec{x}_p$ and $\vec{x}_g$ the corresponding 3D coordinates. This notation also applies hereafter. The regularization terms are: the normal consistency loss

$$\mathcal{L}_{NC} = \frac{1}{|N_F|} \sum_{(f_0,f_1) \in N_F} 1 - \frac{\vec{n}_0 \cdot \vec{n}_1}{||\vec{n}_0|| \cdot ||\vec{n}_1||}, \tag{3}$$

where f_0 and f_1 denote the neighbouring faces in N_F, the set of neighbouring faces of the mesh and $\vec{n}_0$ and $\vec{n}_1$ the normal vectors of the respective neighboring faces; the Laplacian smoothing constraint

$$\gamma_{\mathrm{L}} = \frac{1}{|S_p|} \sum_{v \in S_p} || \frac{1}{|N(v)|} \sum_{v' \in N(v)} (\vec{x} - \vec{x'}) ||, \tag{4}$$

where $N(v)$ is the set of neighboring vertices of vertex v; and the edge length loss

$$\mathcal{L}_E = \frac{1}{|E|} \sum_{(v_0,v_1) \in E} (||\vec{x}_0 - \vec{x}_1||_2 - L_t)^2, \tag{5}$$

where E denotes the set of edges of a mesh. v_0 and v_1 are the vertices composing the edge, and L_t is a constant target length, in our case 0 to ensure edges of similar length.

We introduced an additional *centerline coverage loss* $\mathcal{L}_{CC}$ to enforce the desired branching based on the ground-truth centerlines computed with morphMan [7]:

$$\mathcal{L}_{CC} = \sum_{e \in E_o} d_e, \tag{6}$$

We penalize centerline endpoints located outside of the generated mesh. E_o is the set of centerline endpoints outside of the mesh and d_e is the distance between the endpoint e and the corresponding closest point on the mesh [12].
Thus, our Voxel2Mesh implementation balances between a smooth mesh and the formation of branches. Furthermore, we use a cross-entropy loss $\mathcal{L}_{VC}$ (analog to equation (1)) for the vertex class prediction. The final loss function L is a linear combination of all loss terms $\mathcal{L}$ weighted with their respective hyperparameters λ at different resolutions.

$$L = \lambda_{CE}\mathcal{L}_{CE} + \lambda_{CD}\mathcal{L}_{CD} + \lambda_{NC}\mathcal{L}_{NC} + \lambda_{L}\mathcal{L}_{L} + \lambda_{E}\mathcal{L}_{E} + \lambda_{CC}\mathcal{L}_{CC} + \lambda_{VC}\mathcal{L}_{VC} \tag{7}$$

Based on a grid search with 5-fold cross-validation, the loss hyperparameters were uniformly weighted across all resolutions with: $\lambda_{CE} = 5$, $\lambda_{CD} = 15$, $\lambda_{NC} = 0.1$, $\lambda_{L} = 0.1$, $\lambda_{E} = 0.1$, $\lambda_{CC} = 100$ and $\lambda_{VC} = 5$.
We rely on a uniform upsampling strategy instead of the adaptive upsampling suggested in the original Voxel2Mesh architecture [17], as too many vertices were discarded due to the close adjacency of the branches.
We performed random augmentation during training: affine (rotation -10 to 10°C, scaling 0.9 to 1.1, translation up to 5 mm), blur (standard deviation of the Gaussian Kernel up to 2 mm), and Gaussian noise (mean of 0 and standard deviation up to 0.25 mm).

2.3 Postprocessing

Due to the topology restrictions of Voxel2Mesh, we obtain closed surface meshes, which we need to open for simulation. We approach this by additionally predicting a class for each vertex, defining whether it belongs to an opening or not. We then remove all faces containing opening vertices. The resulting boundary vertices for each opening are then adjusted to lie on the same average plane by least square approximation. At this point, few openings are not detected. Hence, we manually open the remaining closed ends for the subsequent steps.

2.4 Centerline-Based Analysis

For comparison of the anatomical features, we automatically obtained length and diameters for different segments of the PA, as well as the tortuosity of LPA

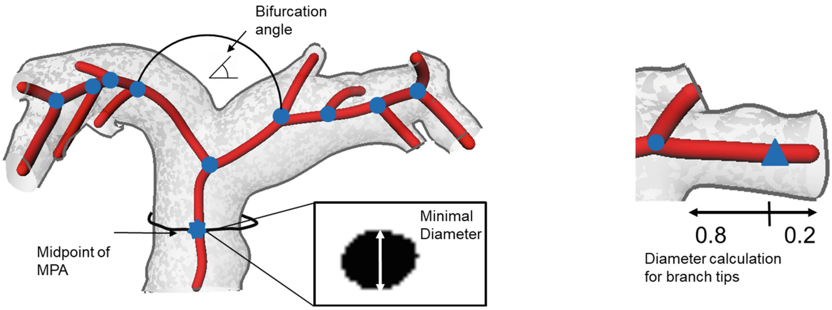

Fig. 3. Anatomical measurements from centerline-based analysis: The centerline (red) is subdivided into several edges, connected by nodes (blue), marking a bifurcation. Diameters for each edge are obtained at the midpoint (blue star) between start and end node of an edge or at start node + 0.8 * (end node - start node) for branch tips (blue triangle). The bifurcation angle between LPA and RPA is the angle between the vectors connecting the endpoint of the MPA to the endpoint of the first edge of LPA and RPA, respectively. (Color figure online)

and RPA and their bifurcation angle from a centerline-based analysis of the open-ended surfaces. The analysis was done using MeVisLab software (MeVis Medical Solutions AG, Germany).

Edges until the first bifurcation were considered to be the MPA. The LPA and RPA were defined depending on the orientation within the scanner, respectively as the longest possible path. All other edges were considered side branches.

For each edge, we evaluate the length as well as the minimal diameter of the surrounding surface. For the MPA and internal edges, the diameters were obtained at the midpoint of each edge, for branch tips closer to the endpoints (see Fig. 3). We accumulated the measurements to calculate the overall length and length-weighted diameters of the MPA, LPA and RPA, respectively. Edges with a length below 10 mm were disregarded on the length-weighted diameter calculation, as side branches will likely distort the diameter measurements. The tortuosity was calculated from the distance of the start and end nodes divided by the overall length of the considered segment. Lastly, the bifurcation angle between LPA and RPA was determined, by calculating the angle between the vectors connecting the endpoint of the MPA to the endpoint of the first edges of LPA and RPA, respectively, with an overall length of at least 10 mm.

2.5 Hemodynamic Simulation

To assess whether the automatically generated surface geometries can be used for the calculation of the pulmonary artery hemodynamics, a steady-state simulation of the peak-systolic state was conducted, as previously described by Brüning et al. [1], using a finite volume solver (STAR-CCM+, v. 17.06, Siemens PLM). Initially, the surface repair functionality of STAR-CCM+ was used to repair any persisting topology errors in the surface geometries, such as non-manifold edges or piercing surface elements. Open edges of the surface mesh were identified automatically and closed off subsequently for boundary condition specification. For all outlets, a zero-pressure boundary condition was specified, whereas a constant mass flow of 350 ml/s, which equaled the average peak-systolic flow rate observed in our sample, was specified as the inlet at the right ventricular outflow tract. A volume mesh was generated using a mesh size of 0.5 mm and a boundary layer consisting of five layers, with an overall thickness of 50 percent of the base size. Simulations were deemed converged when residuals of momentum and mass were below $1e{-}5$.

3 Results

We validated our neural network on 8 cases and evaluated the resulting surface meshes on 10 further cases, which were each processed by two observers.

3.1 Voxel2Mesh - Comparison to Baselines

Training and inference was performed on Dell PowerEdge R7525 compute nodes with 64 AMD Epyc cores, 512 GB RAM and NVIDIA A100 40G or 80G GPU. We used Python 3.9.12 with PyTorch 1.11.0+cu113 and PyTorch3D 0.6.1. As a baseline, we use the state-of-the-art nnU-Net [6], followed by marching cubes (MC) [9] for surface generation. As the vertex class is not directly transferable, we disregard it. Furthermore, we conduct an ablation study, by evaluating the effect of our centerline coverage loss in comparison to a standard Voxel2Mesh.

Figure 4 demonstrates the effect of the centerline coverage loss, which facilitates the formation of the proper branching structure and alleviates the effect of smoothing.

Table 1. Comparative results against baselines on the validation data set of 8 patients using the Intersection over Union (IoU) and mean distance (md), as well as the F1 score for the vertex class predicted by Voxel2Mesh. Our method, Voxel2Mesh + $\mathcal{L}_{CC}$, combines the standard Voxel2Mesh loss functions with an additional centerline coverage loss.

	IoU	md [mm]	vertex class F1
nnU-Net + MC	0.93 ± 0.01	2.9 ± 1.1	N/A
Voxel2Mesh	0.64 ± 0.08	7.3 ± 0.9	0.45 ± 0.28
Voxel2Mesh + $\mathcal{L}_{CC}$	0.87 ± 0.06	2.3 ± 0.6	0.58 ± 0.11

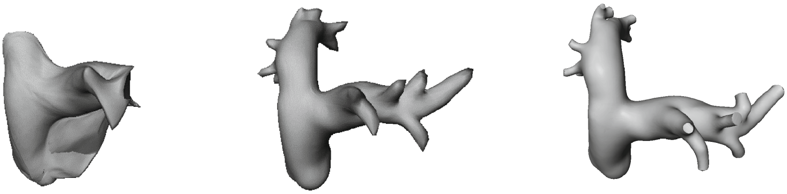

Fig. 4. Example of a surface mesh before (left) and after (middle) introduction of the centerline coverage loss. The target mesh (right) is shown for reference. The centerline coverage loss facilitates the proper branching structure.

3.2 Centerline-Based Analysis of Anatomical Measurements

We performed an inter-observer variability analysis for the 10 CT images annotated by two observers. For this, we manually revised the openings resulting from our Voxel2Mesh algorithm. In the 10 cases, we had to open on average three (23%) additional outlets, that were not detected automatically.
For several measurements, we calculate the mean differences from our reconstructed surface mesh to the average measurements from the two observer reconstructed meshes. We contrast this with the inter-observer difference in Table 2. With a two-tailed paired t-test with a significance level of 0.05, we don't have any significant differences in the shape of the LPA and RPA and the length of the MPA. Differences in the diameter of the MPA ($p \approx 0.023$) and the bifurcation are significant. ($p \approx 0.037$).

Table 2. Centerline-based analysis of anatomical features: We report the mean difference for our automatically generated mesh to the target meshes by the two observers, as well as the inter-observer difference, averaged over all 10 cases.

Measurement		Mean Difference to Target	Inter-observer Difference
MPA	Length [mm]	2.66 ± 1.56	1.89 ± 2.83
	Diameter [mm]	1.17 ± 0.58	0.62 ± 0.56
LPA	Length [mm]	6.69 ± 4.02	8.14 ± 7.18
	Diameter [mm]	1.26 ± 0.66	1.19 ± 0.93
	Tortuosity	0.04 ± 0.03	0.06 ± 0.06
RPA	Length [mm]	5.25 ± 3.49	6.12 ± 4.52
	Diameter [mm]	1.06 ± 0.83	1.23 ± 1.14
	Tortuosity	0.06 ± 0.05	0.07 ± 0.08
Bifurcation [°]		2.38 ± 0.98	1.24 ± 0.65

3.3 Simulation

Simulation of the patient-specific hemodynamics using the automatically generated surface meshes was feasible in an automated manner. Isolated topological errors could be fixed with the automated surface remeshing algorithm used in the numerical pipeline in all instances. No subsequent manual interaction was necessary to process the surface meshes for the numerical simulation. The visualization of resulting intravascular hemodynamics showed a good agreement between flow fields calculated using manually and Voxel2Mesh generated meshes (Fig. 5).

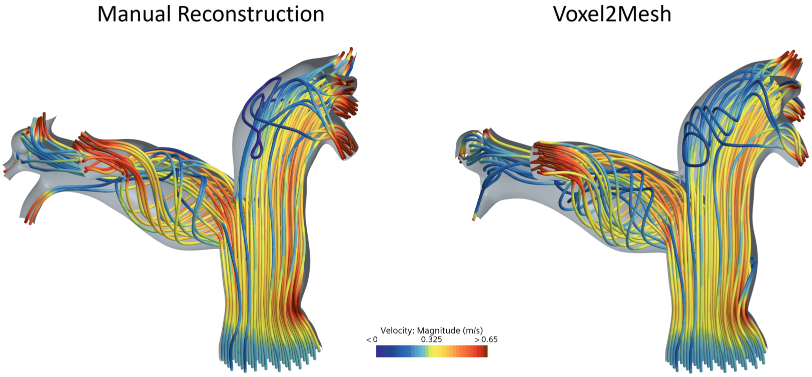

Fig. 5. Exemplary comparison of the simulation of the patient-specific hemodynamics, using the manually and the automatically generated surface mesh, illustrated with velocity-colored streamlines.

Table 3 shows a summary of simulated velocities for the meshes generated by Voxel2Mesh and our two observers. The correlations between our generated meshes and the two observers are comparable.

4 Discussion

We extended the Voxel2Mesh algorithm with a centerline coverage loss to promote pulmonary artery branching and a vertex class prediction for subsequent opening. Derived anatomical measurements from our resulting surface meshes are in accordance with results from two observers. We only see significant differences for the main pulmonary artery and the bifurcation angle. A minor change in the cut-off length of the MPA leads to a change in the measurement position for the diameter and likely also the angle, hence resulting in a significantly different diameter, while the overall shape remains the same. The differences between the bifurcation angles are in the range of 1 to 3°C, so even minor deviations lead to statistically significant differences. However, we do not expect any impact on the hemodynamic simulations from the detected differences in the

Table 3. Velocity measurements from simulations using the mesh from Voxel2Mesh (V2M) and two observers (O1 & O2): We report the mean and standard deviation (Std) for the mean and maximum velocity in meters per second. The correlation between Voxel2Mesh and the observers as well as the inter-observer correlation is shown.

	Mean Velocity [m/s]			Max Velocity [m/s]		
	V2M	O1	O2	V2M	O1	O2
Mean	0.265	0.295	0.312	0.744	0.787	0.844
Std	0.046	0.058	0.070	0.143	0.171	0.232
	V2M/O1	V2M/02	O1/O2	V2M/O1	V2M/02	O1/O2
Correlation	0.981	0.966	0.955	0.882	0.875	0.846

anatomy and the comparison showed a good agreement (Table 3). We experienced self-folding surface meshes, a known issue in surface mesh generation [14], obstructing the hemodynamic simulations. For avoiding such foldings, well-defined training meshes are of utter importance - they need to be watertight, with consistently oriented normals, and also very thin branches should be truncated, to prevent topological errors. Pretraining the segmentation without mesh deformation, reducing the learning rate as well as initially increasing the smoothing factors and slowly reducing them during the training, reduced the chance of self-foldings.

4.1 Time Investment - Manual Vs Automatic Mesh Generation

The manual mesh generation requires four steps: Image-based segmentation of a voxel mask containing the pulmonary artery lumen, generation of an initial surface mesh that is subsequently smoothed, correction of errors in surface topology, and finally truncation of vessel endings. The majority of time necessary for the procedure was spent on the initial segmentation step. Depending on the quality of the image data, this step took between 30 and 180 min per case. Substantial manual effort was required to eradicate fused branches and other artifacts, leading to errors in the deduced meshes (see Table 1). Similarly, smoothing and correction of topological errors of the surface depended strongly on the image quality and thus the quality of the initial surface mesh. This step took between 1 and 10 min. Finally, the truncation of vessel endings took approximately 3 min.

The automated procedure takes approximately 2 s per case for surface reconstruction, with an additional 1 to 3 min of manual interaction for possible error correction, including truncation of remaining vessel endings.

5 Conclusion

We have presented a deep learning-based approach for automatic pulmonary artery mesh generation from CT images with subsequent in- and outlet opening for numerical flow simulation. Apart from the last step of quality assessment and possible error correction, the procedure is fully automated and does not require any human interaction. Therefore, the method enables cost reduction and better reproducibility by minimizing operator-dependent influences. For future work, we consider directly penalizing self-foldings in our loss function or including the automatic removal in our processing pipeline. The nnU-Net demonstrated superior segmentation performance compared to our Voxel2Mesh segmentation pipeline. Thus, directly incorporating the nnU-Net into the Voxel2Mesh architecture might further improve results in the future. Apart from that, an increase in training sample size likely improves the robustness and overall quality of the proposed method. Future work will also focus on enhancing fully automatic outlet opening.

Acknowledgement. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N° 101017578.

References

1. Brüning, J., et al.: In-silico enhanced animal study of pulmonary artery pressure sensors: assessing hemodynamics using computational fluid dynamics. Front. Cardiovasc. Med. **10**, 1193209 (2023). https://doi.org/10.3389/fcvm.2023.1193209
2. Frey, P., George, P.L.: Mesh Generation: Application to Finite Elements, 2nd edn. Wiley, Ltd (2008)
3. Ballester, M.A.G., Zisserman, A.P., Brady, M.: Estimation of the partial volume effect in MRI. Med. Image Anal. **6**(4), 389–405 (2002). https://doi.org/10.1016/S1361-8415(02)00061-0
4. Guo, Q., Gao, C., Liu, M., Wang, H., Yuan, H.: Pulmonary artery segmentation based on three-dimensional region growth approach. In: Zhao, Y., Barnes, N., Chen, B., Westermann, R., Kong, X., Lin, C. (eds.) Image and Graphics, pp. 548–557. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-34110-7_46
5. Hunter, K.S., Feinstein, J.A., Ivy, D.D., Shandas, R.: Computational simulation of the pulmonary arteries and its role in the study of pediatric pulmonary hypertension. Prog. Pediatr. Cardiol. **30**(1), 63–69 (2010). https://doi.org/10.1016/j.ppedcard.2010.09.008, proceedings of the 1st International Conference on Computational Simulation in Congenital Heart Disease
6. Isensee, F., et al.: nnU-Net: self-adapting framework for U-Net-based medical image segmentation. arXiv:1809.10486 (2018)
7. Kjeldsberg, H.A., Bergersen, A.W., Valen-Sendstad, K.: morphMan: automated manipulation of vascular geometries. J. Open Source Softw. **4**(35), 1065 (2019). https://doi.org/10.21105/joss.01065
8. Kong, F., Shadden, S.C.: Whole heart mesh generation for image-based computational simulations by learning free-from deformations. In: de Bruijne, M., et al. (eds.) Medical Image Computing and Computer Assisted Intervention - MICCAI 2021, pp. 550–559. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-87202-1_53
9. Lewiner, T., Lopes, H., Vieira, A.W., Tavares, G.: Efficient implementation of marching cubes' cases with topological guarantees. J. Graph. Tools **8**(2), 1–15 (2003). https://doi.org/10.1080/10867651.2003.10487582

10. López-Linares Román, K., et al.: 3D pulmonary artery segmentation from CTA scans using deep learning with realistic data augmentation. In: Stoyanov, D., et al. (eds.) RAMBO/BIA/TIA -2018. LNCS, vol. 11040, pp. 225–237. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00946-5_23
11. Nguyen, T.D., Kadri, O.E., Voronov, R.S.: An introductory overview of image-based computational modeling in personalized cardiovascular medicine. Front. Bioeng. Biotechnol. **8**, 529365 (2020). https://doi.org/10.3389/fbioe.2020.529365
12. Osher, S., Fedkiw, R.: Level Set Methods and Dynamic Implicit Surfaces. AMS, vol. 153. Springer, New York (2003). https://doi.org/10.1007/b98879
13. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. arXiv:1505.04597 (2015)
14. Sfikas, K., Perakis, P., Theoharis, T.: For^{2}m: recognition and repair of foldings in mesh surfaces. Application to 3D object degradation. arXiv:2206.09699 (2022)
15. Shewchuk, J.: What is a good linear finite element? - interpolation, conditioning, anisotropy, and quality measures. In: Proceedings of the 11th International Meshing Roundtable, vol. 73 (2002)
16. Taylor, C., Figueroa, C.: Patient-specific modeling of cardiovascular mechanics. Annu. Rev. Biomed. Eng. **11**(1), 109–134 (2009). https://doi.org/10.1146/annurev.bioeng.10.061807.160521, pMID: 19400706
17. Wickramasinghe, U., Remelli, E., Knott, G., Fua, P.: Voxel2Mesh: 3D mesh model generation from volumetric data. arXiv:1912.03681 (2019)
18. Zhang, C., Sun, M., Wei, Y., Zhang, H., Xie, S., Liu, T.: Automatic segmentation of arterial tree from 3D computed tomographic pulmonary angiography (CTPA) scans. Comput. Assist. Surg. **24**(sup2), 79–86 (2019). https://doi.org/10.1080/24699322.2019.1649077, pMID: 31401886