Clustering of Aortic Vortex Flow in Cardiac 4D PC-MRI Data

Monique Meuschke[1], Kai Lawonn[1,2], Benjamin Köhler[1], Uta Preim[3], Bernhard Preim[1]

[1]Dept. of Computer Graphics and Simulation, OvG-University Magdeburg, Germany
[2] Institute of Computational Visualistics, University of Koblenz - Landau, Germany
[3]Dept. of Diagnostic Radiology, Municipal Hospital, Magdeburg, Germany
meuschke@isg.cs.uni-magdeburg.de

Abstract. This paper presents a method for clustering aortic vortical blood flow using a reliable dissimilarity measure combined with a clustering technique. Current medical studies investigate specific properties of aberrant blood flow patterns such as vortices, since a correlation to the genesis and evolution of various cardiovascular diseases is assumed. The classification requires a precise definition of spatio-temporal vortex entities, which is performed manually. This task is time-consuming for larger studies and error-prone due to inter-observer variability. In contrast, our method allows an automatic and reliable vortex clustering that facilitates the vortex classification. We introduce an efficient calculation of a dissimilarity measure that groups spatio-temporally adjacent vortices. We combine our dissimilarity measure with the most commonly used clustering techniques. Each combination was applied to 15 4D PC-MRI datasets. The clustering results were qualitatively compared to a manually generated ground truth of two domain experts.

1 Introduction

Cardiovascular diseases (CVDs) are the leading cause of death worldwide. Recent medical studies revealed a strong relation between the occurrence of vortex flow and CVDs [1]. The patient-specific hemodynamic is acquired non-invasively by four-dimensional phase-contrast magnetic resonance imaging (4D PC-MRI) [2]. A qualitative analysis of the obtained blood flow data enables the extraction of vortices. To investigate their influence on pathological genesis such as vascular dilatations, medical studies with homogeneous patient groups are performed to determine in which vessel sections and phase of the cardiac cycle (systole/diastole) vortices occur. Such a vortex analysis is manually performed with common flow visualization techniques such as particle animations. This is a time-consuming and tedious process in which small vortices are easily overlooked. Köhler et al. [3] presented a semi-automatic method based on *line predicates* to filter vortex flow. Nevertheless, spatially and temporally adjacent vortices cause visual clutter and occlusion, which complicates the classification of individual vortices. Therefore, clustering methods are used to separate vortices according to a measure

of dissimilarity. These techniques can be categorized into vector field-based and integral curve-based approaches. Vector field-based methods operate entirely on the measured vector field, with only few systems that can handle the relation between the space and time domains in 3D unsteady flow fields [4]. Integral curve-based approaches partition a set of integral curves by grouping similar curves into clusters [5]. Frequently used dissimilarity measures are spatio-temporal distances or line curvature [6]. All techniques fail to provide meaningful clusters that represent important structures such as vortices.

In this work, we combine the line predicates [7] with different integral curve-based clustering methods to enable an efficient exploration of aortic vortices. A line predicate is a Boolean function that decides if an integral line satisfies a certain property such as scalar values that describe the line geometry. First, line predicates allow to filter vortex flow represented by pathlines. Afterwards, clustering enables to determine the vortex number and provides a reasonable cluster visualization. For this, we applied three well-established clustering techniques. In a qualitative evaluation, we compared the results of healthy and pathological 4D PC-MRI datasets to manually identified clusters of two domain experts.

2 Material and Methods

In this section we describe the acquisition and preprocessing of 4D PC-MRI data, our dissimilarity measure and three clustering methods that we have compared.

2.1 Data Acquisition and Preprocessing

4D PC-MRI data represent one full heartbeat and were acquired using a 3 T Siemens Magnetom Verio MR scanner with a maximum expected velocity (VENC) of 1.5 m/s per dimension. A dataset contains each three (x-, y- and z-direction) time-resolved flow and magnitude images that describe the flow direction and strength, respectively. The spatial-temporal resolution is about $1.77 \times 1.77 \times 3.5$ mm^3 with a 132×192 grid for each of the 15 to 23 slices and 50 ms between the 14 to 21 time steps. In addition to the implementation of an artifact reduction in the flow images, a temporal maximum intensity projection (TMIP) is generated from the magnitude images as basis for a binary segmentation [3]. The vessel surface is extracted via Marching Cubes and then used to extract a centerline that is present as a discrete point sequence with 0.5 mm intervals. Vortex-representing pathlines were extracted semi-automatically using the line predicates technique [3].

2.2 Dissimilarity Measure and Clustering

In the following, we give a detailed explanation of our pathline dissimilarity measure as basis for the clustering. We proceed with a description of three clustering methods that do not need a priori selection of the cluster number, because the number of vortices is often unknown. Each pathline is assigned to exactly one cluster.

Dissimilarity function: The pairwise calculation of pathline dissimilarity includes three distance measures. Based on the observation that vortices are often clearly spatio-temporally separated from each other, Euclidean and temporal distances are considered. Sometimes two vortices show a very similar spatio-temporal behavior, but differ in their distance to the centerline. Therefore, the difference of two lines according to their averaged centerline distance (ACD) was used. The ACD of a line results from the sum of the Euclidean distances of each pathline point to its nearest centerline point. We assume that two pathlines of the same cluster start and end in similar spatial regions and time points. Thus, the spatial and temporal distance comparisons are only necessary between the start and end point of lines, which reduces the computational effort. Each of the five resulting distance values between two given pathlines is normalized to [0,1] using the corresponding maximum value before they are accumulated. The pairwise inter-pathline distances are stored in a squared dissimilarity matrix $\mathbf{M}$. The entry $\mathbf{M}_{ij}$ corresponds to the dissimilarity between pathlines i and j. $\mathbf{M}$ is therefore symmetric, its main diagonal is composed of zeros.

Agglomerative Hierarchical Clustering: The *Agglomerative Hierarchical Clustering* (AHC) builds a cluster hierarchy by initiating each pathline as a cluster. Based on $\mathbf{M}$ and a measure of cluster proximity, the two closest clusters are merged recursively until a single cluster remains. We compare four commonly used proximity measures: *single link*, *complete link*, *average link*, and *Ward's method*. The cluster hierarchy enables a fast analysis of different cluster numbers. Furthermore, AHC is non-parametric, except for $\mathbf{M}$ and the proximity measure. For determination of an appropriate cluster number we use the *L-method* [8].

Spectral Clustering: The idea of *Spectral Clustering* (SC) is to generate a lower dimensional feature space where pathlines are represented as points. This representation improves the separability of clusters and enables the application of a simple clustering technique like *k-means*. The parameter σ controls the cluster similarity and k the number of clusters. Both were calculated automatically using the method suggested in [9].

Density-based clustering: Density-based methods such as the *Density-Based Spatial Clustering of Application with Noise* (DBSCAN) identify dense regions in the feature space. The density of an object o results from the number of objects (including o) in its ϵ-neighborhood. To form a valid cluster, the density of o has to reach a given threshold $minObjects$. We set $minObjects$ to 1, so that each cluster contains at least one pathline. This allows the separation of clusters that differ strongly in their pathline densities. The size of the ϵ-neighborhood relates to the dissimilarity between two objects. We set ϵ to 10 % of the maximum occurring value in $\mathbf{M}$. A smaller ϵ value led to very small clusters, which requires a manual cluster merging to reach the ground truth. With greater ϵ values the clusters were too large so that the experts' results cannot be replicated.

2.3 Evaluation

We used 15 datasets for the evaluation: 2 healthy volunteers with a slight physiologic helix in the aortic arch and 13 patients with different CVDs. Each patient has prominent vortex flow in different parts of the aorta. Our clustering framework differentiates at most ten clusters via color. More is not necessary because no more than four vortices occur in our employed datasets. Thus, the remaining clusters can be used to exclude laminar flow or noise from vortex analysis. In addition to the possibility of deselecting clusters the user can merge clusters or change the cluster number if the automatically calculated number is not appropriate. We investigate which of the three clustering methods combined with our dissimilarity measure produces the best qualitative clustering results. For this, we compare the clustering results to a manually generated ground truth of our collaborating experts by counting the necessary interactions (merge, change number) to reproduce the experts' results. One to three steps were considered acceptable. The first expert is a cardiologist with four years of work experience in the analysis of 4D PC-MRI datasets. In addition, an expert specialized in the visualization of 4D PC-MRI data was involved. The line predicate-based vortex filtering was applied in two varying degrees to evaluate the robustness of our dissimilarity measure. The first one produces a strongly filtered set of pathlines, whereas the second one contains more laminar flow and noise.

3 Results

In Fig. 1 the clustering results of five patients with different CVDs are shown. The AHC results are illustrated in the first row with a preview of the original pathlines on the bottom left. The results of SC and DBSCAN are presented in the second and third row, respectively. The computation time per case is between 2 and 9 s, depending on the amount of pathlines. Our testing system uses an Intel Core i7 CPU with 2 GHz, 12 GB RAM and a NVidia GeForce GT 540M. The first example (Fig. 1a) shows stronger filtered vortex flow containing two clearly separated vortices. Such cases could be clustered automatically with all three methods (Fig. 1a, f, k). In the second and third example (Fig. 1b-c) the vortices strongly differ in their pathline density and are partly adjacent in space and time, which can be handled reliably by AHC. SC was not able to deal with widely varying cluster densities (Fig. 1g), whereas DBSCAN can often not cluster spatio-temporal vortices (Fig. 1m). For all stronger filtered datasets AHC could generate the experts' results automatically in nine cases, and for the remaining six cases three merging steps were at most necessary. DBSCAN and SC failed in 2 datasets because it was not possible to produce the experts' clustering automatically or using the cluster interactions. Finally, we applied the clustering methods to the slightly filtered version of the datasets illustrated by two examples in (Fig. 1d-e). In this examples we hide clusters that represent laminar flow or noise. In 13 cases, three merging steps were at most necessary with AHC and more than three in the other two datasets. In 11 cases, SC produced similar good results as AHC (Fig. 1i) but in 4 cases it was not possible to replicate the experts'

results. The clusters were too large and contained laminar flow and noise (Fig. 1j). DBSCAN could generate the experts' results with more than three merging steps in six datasets but failed in nine cases (Fig. 1n, o).

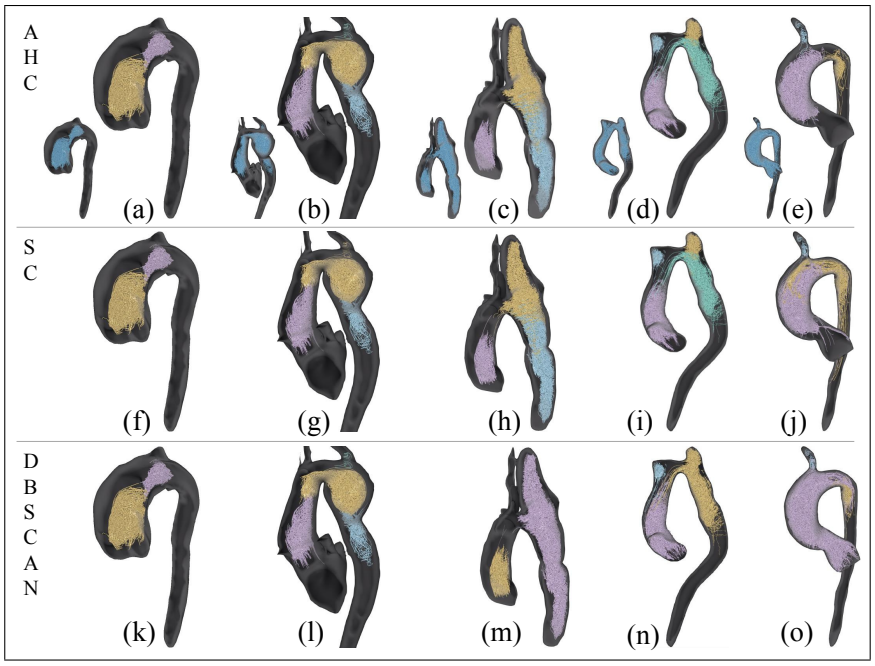

Fig. 1. Exemplary clustering results for five patients. The first row (a-e) shows the AHC results together with a thumbnail of the original pathlines. The results of the SC and DBSCAN are presented in the second and third row. The combination of AHC and our dissimilarity measure is able to group well separated vortices (a) and roughly filtered datasets (d-e). Moreover, we can cluster vortices with strongly varying pathline densities (b) as well as spatially and temporally adjacent vortices (c). Such characteristics are often difficult to handle with the other methods (g, j, m-o).

4 Discussion

We presented a method for clustering vortical blood flow in the aorta. A comparison of three clustering methods showed that AHC using average link was the only one that results in correct clusters for all datasets comparing to the ground truth. Both strongly and slightly filtered vortex flow was separated reliably. Our method facilitates an easier classification of vortices. Domain experts could determine their positions and temporal behavior by the use of a pathline animation. Furthermore, they can incorporate their knowledge by changing the calculated cluster number and merge or deselect single clusters. Thus, the expert

has control over the results and is able to handle anatomical diversities such as various vortex numbers and sizes. A possible application of our method is to classify aortic vortex flow in larger studies. For example, patients with a bicuspid aortic valve show different flow patterns depending on which of the three cusps are merged. Our method supports an easier determination of specific vortex properties such as the vortex shape, size and location which is important to understand their influence on pathological vascular changes. In addition, our technique will in future form the basis for an automatic vortex classification. Currently, our framework hides clusters that represent laminar flow or noise not automatically. This needs to be enhanced in the future. Moreover, AHC requires more than three merging steps in few cases which can potentially be improved by using methods like the *Ensemble Clustering* [10]. A topic for future work is the adaption of clustering in branching vessels like the pulmonary artery or in adjacent areas like the ventricles.

5 Acknowledgements

This work was partially funded by the BMBF (STIMULATE-OVGU:13GW0095A). The authors like to thank the Heart Center Leipzig for providing us the datasets.

References

1. Hope MD, Wrenn J, Sigovan M, et al. Imaging Biomarkers of Aortic Disease - Increased Growth Rates with Eccentric Systolic Flow. J Amer Coll Cardiol. 2012;60:356–7.
2. Markl M, Frydrychowicz A, Kozerke S, et al. 4D Flow MRI. J Magn Reson Imaging. 2012;36:1015–36.
3. Köhler B, Gasteiger R, Preim U, et al. Semi-Automatic Vortex Extraction in 4D PC-MRI Cardiac Blood Flow Data using Line Predicates. IEEE Trans Vis Comput Graph. 2013;19(12):2773–82.
4. van Pelt RFP, Sander J, ter Haar Romeny B, et al. Visualization of 4D Blood-Flow Fields by Spatiotemporal Hierarchical Clustering. Comput Graph Forum. 2012;31(3pt2):1065–74.
5. Oeltze S, Lehmann DJ, Kuhn A, et al. Blood Flow Clustering and Applications in Virtual Stenting of Intracranial Aneurysms. IEEE Trans Vis Comput Graph. 2014;20(5):686–701.
6. McLoughlin T, Jones MW, Laramee RS, et al. Similarity Measures for Enhancing Interactive Streamline Seeding. IEEE Trans Vis Comput Graph. 2013;19(8):1342–53.
7. Salzbrunn T, Garth C, Scheuermann G, et al. Pathline Predicates and Unsteady Flow Structures. The Visual Computer. 2008;24(12):1039–51.
8. Salvador S, Chan P. Determining the Number of Clusters/Segments in Hierarchical Clustering/Segmentation Algorithms. In: IEEE International Conference on Tools with Artificial Intelligence; 2004. p. 576–84.
9. Zelnik-Manor L, Perona P. Self-tuning Spectral Clustering. In: Advances in Neural Information Processing Systems 17. MIT Press; 2004. p. 1601–8.
10. Strehl A, Ghosh J. Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. J Mach Learn Res. 2002;3:583–617.