# Robust Vessel Path Tracking in Imperfect Volumetric Data

N. C. Codella[1], Y. Wang[2], R. Zabih[3]

[1]Physiology, Biophysics, and Systems Biology, Cornell Weill Medical College, New York, NY, United States, [2]Cornell Weill Medical College, New York, NY, United States, [3]Computer Science, Cornell University, Ithaca, NY, United States

## INTRODUCTION

Many algorithms exist that analyze the properties of an artery visualized from angiographic images. Some algorithms measure artery dimensions, but require a user to specify points that define the complete axial path of the artery (2). Others employ region-growing techniques that determine the path automatically (3). The former requires a significant amount of user interaction, which is time consuming. The latter may not function properly if there is a loss of contrast along one edge of the artery. In this work we present a fast and efficient algorithm that can determine the path of an artery of any length, visualized in 3D angiography data with low SNR and loss of contrast along one side of the artery from neighboring structures, while relying on minimal user input. Arterial dimensions can be simultaneously recorded during program execution. Running time is on the order of 1-10 seconds on a 1.7 GHz Pentium M. The required user input consists of two points that define the initial arterial axial vector, and a tracking length limit. A fixed number of other inputs controlling edge detection and scan resolution also exist, but are optional. The small amount of user input in this algorithm maintains good reproducibility of results and reduces user labor. Furthermore, we have confirmed through experimentation that slight variation in user input for a given analysis has limited effects on that analysis.

## ALGORITHM DETAILS:

The algorithm consists of three main components: a customized edge detector, a cross-section midpoint calculator, and a path-tracking algorithm. The edge detector is a subcomponent of the midpoint calculator, which is a subcomponent of the tracking algorithm. 3D volumetric data is provided to the software as a series of 2D slices through the volume. Values between voxels are linearly interpolated.

The path-tracking algorithm uses cross-section midpoints to determine continuing vessel direction. At each iteration (cross-section) $c_i$, current vessel direction is represented as a vector, $\boldsymbol{v}_i$, true midpoint as a point $p_i$, and the initial estimated midpoint as a point $e_i$. The estimated midpoint $e_i$ is calculated by moving a unit step of fixed length (axial resolution) from $p_{i-1}$ along $v_{i-1}$. The midpoint calculator then uses $e_i$ to find $p_i$. The current direction $\boldsymbol{v}_i$ is then temporarily defined by the difference between $p_i$ and $p_{i-1}$. A scaling factor $s_i$, inversely proportional to the difference between $\boldsymbol{v}_i$ and $\boldsymbol{v}_{i-1}$ [$s_i = 1 / (1 + |\boldsymbol{v}_i - \boldsymbol{v}_{i-1}|)^{1.2}$] is used to average $\boldsymbol{v}_i$ between itself and $\boldsymbol{v}_{i-1}$: [$\boldsymbol{v}_i = s_i * \boldsymbol{v}_i + (1 - s_i)*\boldsymbol{v}_{i-1}$]. This is done to provide smoothness to counteract noisy data, based on the assumption that vessels do not make abrupt turns. The difference between current and previous direction vectors is then added, scaled by the same factor: [$\boldsymbol{v}_i$ += $s_i * (\boldsymbol{v}_i - \boldsymbol{v}_{i-1})$]. This is done to predict how the vessel's axis is changing, and at the same time increase tolerance to noisy data by summing only a normalized magnitude. The resulting vector is then normalized, and used to estimate the next cross-section midpoint, $e_{i+1}$.

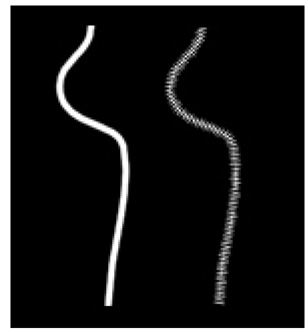

**Figure 1: A single slice through artificial test data (left), and the results of the tracking software (right). Individual cross section scan marks are visible with the axial resolution set to 3 pixel lengths. These results demonstrate the validity of the mathematical models used to construct the tracking algorithm.**

The cross section midpoint calculator performs a set of radial scans originating from the estimated midpoint $e_i$ out to the edge of the vessel wall. Each scan from the estimated midpoint to the vessel wall defines a vector, $\boldsymbol{r}_i$. A center-of-mass approach is used to find the true midpoint: each radius is treated as an isosceles triangle with a mass ($mass_i = area = |\boldsymbol{r}_i|^2$), and a center of mass ($COM_i = 2*\boldsymbol{r}_i/3$). Each [$COM_i*mass_i$] pair is vector summed, and the resulting vector is then divided by the total mass. This final vector gives the vector difference between the initial estimated midpoint and the true midpoint of that cross-section.

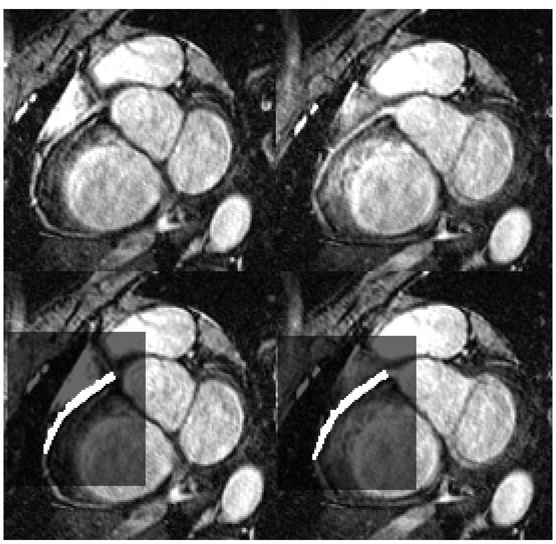

**Figure 2: Input data set (top row) and output data set (bottom row). The software presents output as an overlaid white highlight, as well as a white outline on a black background, both of which have been superimposed. Images are 2D slices through 3D volumetric data. Algorithm tracking is visible as the white highlighted area along the coronary artery. These results show the capability of the algorithm to handle loss of contrast on one side of the vessel, as well as the tracker's ability to deal with other artifacts along the vessel path.**

The edge detector is based on a customized Laplacian algorithm, and incorporates 2 additional basic assumptions: the vessel diameter at the current cross section cannot be significantly wider than the average diameter of the previous cross sections, and the variance of diameter lengths at a given cross section is limited to a certain amount. Such working assumptions lead to a tracking algorithm that functions similar to a driver who follows the road: if one edge of the road (artery wall) becomes invisible (loss of contrast), the driver (tracker) bases the car's position on the visible side of the road (opposite side of artery wall). The width of the lane (artery diameter) is defined as a limited average of the previous lane (artery) widths. Using that diameter value, the other invisible side of the road (artery) can be estimated. The effects of this approach are profound on imperfect data: even though contrast is lost on one side of the artery, the algorithm continues to properly follow the path (Fig. 2).

## DISCUSSION:

The main concept behind the many assumptions applied in the algorithm is that a segment of artery is fairly consistent in its form, and that if any change in the diameter is to occur, it should be narrowing or a quick widening after a point of a stenosis. The application of this concept to the different components of the software leads to extraordinary results: not only does the path tracker function flawlessly in the case of ideally clean volumetric data (Fig. 1), but it also functions well in noisy low resolution data that still has the artery lumen identifiable to the human eye (Fig. 2). The only drawback of the algorithm's assumptions is that branches will be treated as regions of the artery wall that have lost boundary contrast.

Since this algorithm runs instantaneously on image data, it can be used for immediate arterial analysis after image acquisition. During the entire course of path tracking, vessel radius and diameters can be recorded and used to calculate properties of blood flow, which may assist in the diagnosis of a patient's condition. This will be discussed in the full-length paper.

REFERENCES: 1. de Konig PJH et al. MRM 2003:50:1189-1198 2. Frangi AF et al. MMBIA 2000:p.110 3. Yi J et al. IST 2003;13:208-214.