Uncluttered Single-Image Visualization of Vascular Structures Using GPU and Integer Programming

Joong-Ho Won, *Member, IEEE*, Yongkweon Jeon, Jarrett K. Rosenberg,
Sungroh Yoon, *Senior Member, IEEE*, Geoffrey D. Rubin, and Sandy Napel

Abstract—Direct projection of 3D branching structures, such as networks of cables, blood vessels, or neurons onto a 2D image creates the illusion of intersecting structural parts and creates challenges for understanding and communication. We present a method for visualizing such structures, and demonstrate its utility in visualizing the abdominal aorta and its branches, whose tomographic images might be obtained by computed tomography or magnetic resonance angiography, in a single 2D stylistic image, without overlaps among branches. The visualization method, termed uncluttered single-image visualization (USIV), involves optimization of geometry. This paper proposes a novel optimization technique that utilizes an interesting connection of the optimization problem regarding USIV to the protein structure prediction problem. Adopting the integer linear programming-based formulation for the protein structure prediction problem, we tested the proposed technique using 30 visualizations produced from five patient scans with representative anatomical variants in the abdominal aortic vessel tree. The novel technique can exploit commodity-level parallelism, enabling use of general-purpose graphics processing unit (GPGPU) technology that yields a significant speedup. Comparison of the results with the other optimization technique previously reported elsewhere suggests that, in most aspects, the quality of the visualization is comparable to that of the previous one, with a significant gain in the computation time of the algorithm.

Index Terms—Single-image visualization, abdominal aorta, side-chain placement, integer linear programming, GPGPU, CUDA, parallelization

✦

1 INTRODUCTION

THIS paper describes a method for an uncluttered single-image visualization (USIV) of 3D branching structures. Introduced in [1], USIV presents all branches in a single image without misleading overlap, with minimal distortion, thus preventing unnecessary user interaction and enhancing the readability of the 2D visualization. In addition, USIV makes the visualization independent of user interface devices. The output visualization can be statically displayed on a computer screen or even printed on a sheet of paper.

While completely general, USIV was developed as a solution to the visualization of the abdominal aorta and its branches, as they might be captured by computed tomography or magnetic resonance angiography (CTA or MRA).

- *J.-H. Won is with the School of Industrial Management Engineering, Korea University, Engineering Bldg., Anam-dong, Seongbuk-gu, Seoul 136-713, Korea. E-mail: wonj@korea.ac.kr.*
- *Y. Jeon and S. Yoon are with the Department of Electrical and Computer Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151-744, Korea. E-mail: yonkham@acl.snu.ac.kr, sryoon@snu.ac.kr.*
- *J.K. Rosenberg is with the Department of Radiology, Stanford University, Lucas MRS Imaging Center, Room P-284, 1201 Welch Road, Stanford, CA 94305-5488. E-mail: jarrett.rosenberg@stanford.edu.*
- *G.D. Rubin is with the Department of Radiology, Duke University, DUMC 3808, Durham, NC 27710. E-mail: grubin@duke.edu.*
- *S. Napel is with the Department of Radiology, Stanford University, James H. Clark Center, 318 Campus Drive, Room S323, Stanford, CA 94305-5450. E-mail: snapel@stanford.edu.*

In diagnosing and reporting vascular abnormalities from these examinations, it is often desirable to visualize all clinically relevant branches at once. A single schematic image that summarizes the key attributes of a subject's condition can provide a useful map to more complex anatomical and physiological information, and can be made available in a multimedia medical record that incorporates all disparate data into a single view. This type of summary image is analogous to one that is often hand drawn by physicians in medical charts to convey a schematic representation of the subject's anatomy. Rather than serving for primary interpretation, the images communicate key features efficiently through annotation of the drawing. One characteristic of the hand-drawn summary is that it uses an approximately fixed direction by which to view the 3D anatomy. (We refer to this direction as a canonical viewing direction.) Another key characteristic is that the hand-drawn summary allows modification of the spatial configuration of vessel branches in order to reveal as many relevant branches as possible, without hampering the reader's recognition of the anatomy. Conventional computerized summary visualization methods, such as volume rendering (VR) and maximum intensity projection (MIP), rely on simplistic orthogonal or perspective projection so that the 3D form can be displayed on computer screens. This often causes false intersections of vessels that mislead the reader's recognition to appear. Therefore, they are not as effective as a "well-drawn" single, static image. To address the problem of misleading overlaps, they require user's intervention to try several projection directions in turn, or to examine only a small portion at a time. More

advanced "detail and context" methods such as ExoVis [2] that provide a 3D view in the center of the display with translated 2D slice views surrounding it in close proximity are not suitable either: a 3D view inherits the problem of misleading visual clutters of VR; 2D slice views will only show small cross sections of the vessels scattered on the image plane. In general, we can state that the inherent tortuosity of the abdominal aortic vessel tree causes linear visualization methods to fail to reveal the entirety of the vasculature. If all clinically relevant vessels can be shown without overlaps, the resulting single, static image based on a canonical viewing direction will amount to a "well-drawn" schematic representation, and be more effective than VR or MIP in the context of communicating key features such as the branch with an aneurysm. This idea motivates the development of USIV. The evaluation conducted in [1] provides an evidence that USIV is capable of producing images that are recognizable as a map of the underlying anatomy of the abdominal aortic vessel tree.

This paper investigates the connection between USIV and side-chain placement (SCP), an important concept in the protein folding research in computational biology. Using this connection, we adopt a method developed for SCP to USIV: the minimization of the score function, solved using simulated annealing (SA) in [1], is reformulated as an integer linear programming (ILP) over precomputed samples from the optimization search space. The precomputation can be massively parallelized using graphics processing units (GPUs), achieving a significant performance gain over [1]. This precomputation method for the ILP, together with the identification of the connection between USIV and SCP, constitutes the main contribution of the paper.

The paper is organized as follows: The next section discusses related work. Section 4 formulates the USIV optimization problem as an ILP problem, inspired by the analogous formulation of SCP. Section 5 explains parallelization of the precomputation using GPUs. Section 6 describes the test data and the evaluation method. Section 7 presents the results of the evaluation. Discussion and concluding remarks follow thereafter.

2 RELATED WORK

Prior work closely related to USIV is that of Kanitsar et al. [3]. Their method is developed in the context of the curved planar reformation (CPR) visualization technique. CPR can visualize the lumen of a blood vessel on a 2D plane [4]. Largely, because of the difficulty of defining a curved plane spanning multiple branches of vessels, it has been traditionally applied to a single vessel segment only. The untangled CPR proposed in [3] extends the CPR technique to multiple vessels by compositing individual CPRs in such a way that vessel centerlines are hierarchically enclosed by circular sectors and are rotated in order to avoid overlaps among the enclosures. Although the visualization of lumen is not a primary goal of ours, the methodology of [3] provides an important insight into how to handle vascular geometry: the use of the enclosures of vessel centerlines, and the relaxation of spatial coherency. This insight leads to modeling the abdominal aortic vessel tree as an articulated object in USIV.

The articulated object model has been used in the human posture estimation problem [5], [6], [7] studied in the computer vision community. In the cited literature, body parts (e.g., head, torso, arms, legs) are treated as rigid objects, and kinematic constraints are imposed with rotational degrees of freedom. Occlusion constraints, such as that the left hand is partially hidden by the right hand, are imposed by augmenting the graph (usually tree) of kinematic constraints. As stated, USIV adopts the articulated object model to represent the anatomy of the abdominal aortic vessel tree.

Several publications from the visualization community concern illustrative visualization in general. Agrawala et al. studied automatic generation of exploded view diagrams [8], [9], [10], primarily for visualizing CAD models and complicated mathematical surfaces with a single linear explosion axis. Although the surface of the abdominal aortic tree can be viewed as a composition of generalized cylinders, each of which has a curved axis of symmetry, the difficulties of producing an exploded view diagram of a surface having even a single curved axis of symmetry are already pointed out in [8]. The work by Ware and Mitchell [11], [12] is on visualizing graphs by embedding them in the 3D space and providing 3D perception using stereoscopic displays and/or kinetic depth cues. This is the opposite approach to USIV, which tries to embed trees in the 2D plane with hard geometric constraints. In addition, the evaluation in [11] was conducted on random graphs that do not convey an anatomical meaning, and the use of stereoscopic displays that require a fixed eye position may diminish its usability as a tool in clinical context.

More medically oriented publications deal with flattening/unfolding of anatomical structures. Bartrolí et al. [13] present unfolding of the surface of the colon, which has no branching structure, based on centerline extraction and nonuniform raycasting. Zhu et al. [14] present an angle-preserving flattening technique for branching vessel surfaces, and an area-preserving technique, utilizing Dirichlet functionals and optimal mass transport theory. The notion of flattening in these methods is basically tearing a closed-3D surface and unfold it on a plane, and is different in its goal from ours that is rather close to "unrolling" of centerlines. The work by Termeer et al. [15] provides a diagnostic summary of late-enhancement myocardial MR imaging. They introduce the volumetric bulls eye plot (VBEP) annotated with anatomical information. While the VBEP adds the coronary anatomical context to the conventional bull's eye plot, developed due to the insufficient resolution of the coronary arteries in whole heart MRI scans for accurate diameter measurements, in order to assess size and location of infarcted myocardial tissue in spatial relation to the coronary anatomy, a 3D volumetric visualization has to be employed. Ropinski et al. [16] introduce a system for visual analysis of PET/CT scans of the aortic arch of the mouse. The primary purpose of this system is visual comparison of imaged plaque regions in different species in order to support a quantitative in-depth analysis. In contrast, USIV aims to facilitate communicating salient clinical features in an informal but effective manner through a stylistic visualization. This is conceptually similar to the
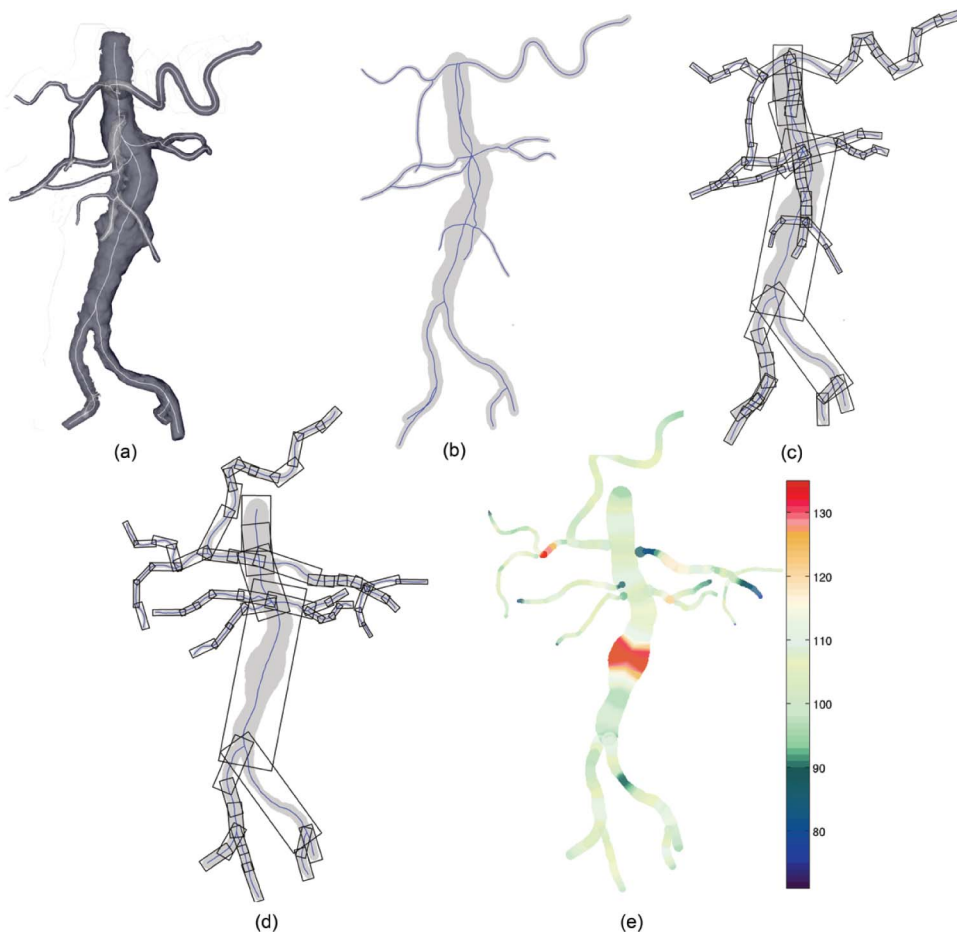
Fig. 1. Graphical illustration of the algorithm. (a) Surface rendering of the abdominal aortic vessel tree together with the 3D centerlines. (b) Two-dimensional centerlines of this vessel tree and the associated vessel diameters. (c) Bounding box representations of the vessel tree. (d) Uncluttered bounding box configuration after minimization of the score function. The 2D vessel centerlines and diameters are transformed according to the new bounding box configuration. (e) Stylistic visualization of (d). The vessel segments are drawn with widths proportional to the diameters, and color coded by the percentage relative to the median diameter of each segment.

work of Oeltze and Preim [17], which models the vascular wall with convolution surfaces, and of Hahn et al. [18], which uses truncated cones to represent schematically the vessels of the liver. USIV, however, focuses on producing a visualization with 2D geometry rather than 3D surfaces.

3 UNCLUTTERED SINGLE-IMAGE VISUALIZATION

Fig. 1 illustrates graphically the steps of USIV, described in the following sections.

3.1 Preprocessing

The first step models the 3D geometry of each vessel segment in the abdominal aortic vessel tree with a tube comprised of its centerline and the diameters along its length. This step can be performed using existing methods, e.g., level-set and tree extraction [19], [20]. These methods can automatically model the topology of multiple branches, and do not require further knowledge of the vascular anatomy, such as the names of individual vessels. The outputs of the modeling process include 3D vessel centerlines and diameters, which are used in subsequent steps (Fig. 1a).

The 3D centerlines obtained in the previous step are collectively "unrolled" using multipath CPR [3] without lumen information, for a canonical viewing direction (usually anteroposterior). The resulting 2D tree retains the diameter information from the 3D geometric model but may exhibit intersections of the projected vessel segment centerlines (Fig. 1b).

For each vessel segment of the 2D representation of the vessel tree, the flattened tube comprised of the flattened centerline and the diameters along its length is simplified by a bounding box aligned with the vector connecting the proximal and the distal points of the flattened centerline. The height of the bounding box is defined to be the length of this vector plus the radii associated with the two endpoints of the flattened centerline, and its width to be the sum of the farthest distances of the centerline from the vector on each side. In order to approximate the tortuous geometry of a vessel segment, if the aspect ratio (width/length) of the bounding box is beyond a threshold, the corresponding vessel segment is subdivided in the middle and new bounding boxes are computed. The orientation Θ, of the bounding box is defined to be the orientation of the height vector with respect to a common coordinate system. The orientation, together with its width, height, and location, fully describes the geometry of the bounding box (Fig. 2). By fixing the width and the height, we can consider the bounding box a rigid part and the 2D vessel tree an articulated object
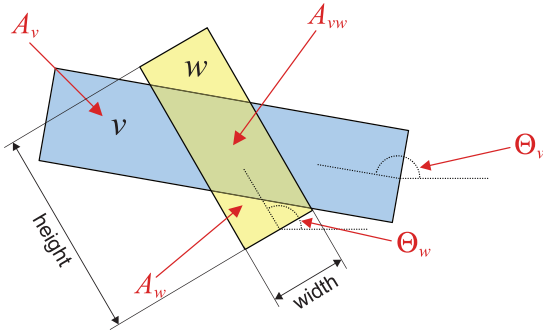
Fig. 2. Illustration of a configuration of a pair of bounding boxes. Notations are described in the text.

comprised of rectangles. We reserve the term *vessel part* in order to refer to the bounding box that encloses a vessel segment. The configuration of the articulated object model is defined to be the orientations of all vessel parts in the model. The initial configuration, given by the centerline flattening in the previous step and the flattened tubes in this step, is referred to as the *reference configuration* with respect to which deformed configurations are evaluated (Fig. 1c).

Note that this modeling procedure does not require knowledge of the anatomy of the vessel tree, other than the identification of the vessel part that corresponds to the root.

3.2 Construction of the Score Function

The score function evaluating a configuration of vessel parts is defined as follows:

$$\mathcal{E}(\Theta_1, \ldots, \Theta_{|V|}) = \Omega(\Theta_1, \ldots, \Theta_{|V|}) + \lambda \cdot \Delta(\Theta_1, \ldots, \Theta_{|V|}), \quad (1)$$

where $\Omega(\cdot)$ is the total overlap (see Section 3.2.1) and $\Delta(\cdot)$ is the total deviation (see Section 3.2.2) of the given configuration, represented by the vector $(\Theta_1, \ldots, \Theta_{|V|})$ of the orientations of the vessel parts. The set V denotes the index set (vertex set) of the vessel parts. The size of V is denoted by $|V|$. The Lagrangian multiplier λ is the weight given to the total deviation.

This formulation encodes our objective of finding a configuration of the vessel parts that 1) minimizes overlap among bounding boxes, and 2) minimizes the deviation of the visualization from the reference configuration. Obviously, unless the reference configuration is free of overlap, these two goals conflict. The quantity λ trades off these conflicting objectives.

3.2.1 Overlap Metrics

The total overlap of the given configuration is defined to be the sum of pairwise overlap metrics over all pairs of vessel parts

$$\Omega(\Theta_1, \ldots, \Theta_{|V|}) = \sum_{(v,w) \notin J} \rho(\Theta_v, \Theta_w) + \alpha \cdot \sum_{(v,w) \in J} \tilde{\rho}(\Theta_v, \Theta_w), \quad (2)$$

where $0 < \alpha < 1$, and the pairwise overlap metrics are given as

$$\rho(\Theta_v, \Theta_w) = \max(A_{vw}(\Theta_v, \Theta_w)/A_v, A_{vw}(\Theta_v, \Theta_w)/A_w)$$
$$= A_{vw}(\Theta_v, \Theta_w)/\min(A_v, A_w), \quad (3)$$

$$\tilde{\rho}(\Theta_v, \Theta_w) = \begin{cases} 0, & 0 \leq \rho(\Theta_v, \Theta_w) < a, \\ \dfrac{1}{b-a}(\rho(\Theta_v, \Theta_w) - a), & \leq \rho(\Theta_v, \Theta_w) < b, \\ 1, & b \leq \rho(\Theta_v, \Theta_w) \leq 1, \end{cases} \quad (4)$$

with $0 \leq a < b \leq 1$. Here, J is a set of joint-sharing vessel part pairs. Equation (3) quantifies the overlap between a pair of vessel segments. For any vessel part pair (v, w), the areas of two bounding boxes and their intersection are denoted by A_v, A_w, and A_{vw}, respectively (Fig. 2). Then, (3) is the larger of the two ratios of the intersection area to the vessel part area. There are unavoidable overlaps at the joints, and (4) is designed so that such overlaps are less penalized than avoidable overlap among nonjoint-sharing pairs; parameters a and b control the reduction in penalty.

3.2.2 Deviation Metrics

The total deviation measures the deviation of a given configuration from the reference configuration as the sum of pairwise deviation metrics for all pairs of joint-sharing vessel parts

$$\Delta(\Theta_1, \ldots, \Theta_{|V|}) = \sum_{(v,w) \in J} \delta(\Theta_v, \Theta_w), \quad (5)$$

where the pairwise deviation metric is given as

$$\delta(\Theta_v, \Theta_w) = \left\| \begin{pmatrix} \cos(\Theta_v - \Theta_w) \\ \sin(\Theta_v - \Theta_w) \end{pmatrix} - \begin{pmatrix} \cos(\Theta_v^{(\text{ref})} - \Theta_w^{(\text{ref})}) \\ \sin(\Theta_v^{(\text{ref})} - \Theta_w^{(\text{ref})}) \end{pmatrix} \right\|^2, \quad (6)$$

where $\Theta_i^{(\text{ref})}$ denotes the orientation of vessel part i in the reference configuration, $i \in V$. Equation (6) is designed so that the sum of the metrics over all joint-sharing vessel part pairs is minimized (zero) when there is no change in their relative orientation $(\Theta_v - \Theta_w)$ from the reference configuration.

3.2.3 Minimization of the Score Function

The task of implementing USIV of the abdominal aortic vessel tree can now be cast as an optimization problem that minimizes the score function (1) over the configuration space of $(\Theta_1, \ldots, \Theta_{|V|})$. Because of the large number of interaction terms and the nonlinearity of each term, minimization of the score function (1) is a hard optimization problem. Intuitively, the problem is difficult because resolving an overlap involving a vessel part pair may introduce a new overlap between other pairs of vessel parts.

One approach to simplify the problem is to reduce the search space. As the largest artery of the body, the aorta can be considered a frame of reference in relation to which other branches are recognized. The common iliac bifurcation serves a similar role, as it is the most noticeable bifurcation in the abdominal aortic vessel tree. Anchoring the vessel parts that correspond to the aorta and common iliac arteries creates a configuration that does not differ much from what is expected from the model anatomy, while reducing the dimension of the problem variables. We call this set of anchored vessel parts the "backbone" of the abdominal aortic vessel tree, for the reason that will become clear in Section 4.

3.3 Reconstruction and Visualization

From the optimized configuration that is expected to be free of branchwise overlaps while minimizing the distortion (Fig. 1d), we reconstruct the stretched vessel centerlines and the diameters along their lengths, and produce a stylistic visualization of the resulting tube model (Fig. 1e).

4 OPTIMIZATION

We begin this section with a brief introduction to the SCP problem and its terminology, explore its connection to USIV, and formulate the mathematical optimization problem using the SCP terminology.

4.1 Formulation Inspired by the Protein Side-Chain Placement Problem

Side-chain placement is an important task in protein structure prediction, which aims to infer a protein's 3D geometrical configuration from its amino acid sequence [21]. A protein is a sequence of amino acids which, under suitable conditions, folds in 3D space to form a geometrical structure. An amino acid consists of a peptide backbone and a group of atoms called a *side-chain*. The side-chain is attached to the peptide backbone and distinguishes one amino acid from the others [22]. The succession of the peptide backbones in an amino acid sequence is called the *backbone* of the protein defined by the sequence. An amino acid block in the protein is called a *residue*. Placement, or finding the best conformation of the side-chains, is important because prediction of the 3D structure of a protein is often limited to the backbone configuration [23]. The *conformation* of the side-chains remains to be found. The conformation of a side-chain is determined by the dihedral angles, denoted by χ_1, χ_2, etc., among the atoms in the chain. SCP is posed as an optimization problem in which the energy of the tertiary structure is defined as a function of the χs, and the global minimum energy conformation is sought. The energy function consists of the covalent terms that evaluate bond angles and torsion among the bonded atoms, and the Van der Waals potential terms that evaluate the distance or collision among any pair of atoms.

Side-chain dihedral angles tend to be clustered around a small set of values of χ, which in general corresponds to the local minima of the potential energy of the isolated amino acid [24]. Such a conformation is called a *rotamer*. That there are only a few rotamers for each amino acid, or residue type, enables discretizing the search space for minimizing the energy function. A popular family of approaches involves *rotamer libraries*. For each residue type, a rotamer library is built by determining the side-chain conformations and their frequencies in well-determined protein structures, e.g., from the protein data bank [25].

Interestingly, USIV bears a very similar mathematical structure to SCP. The immovable vessel parts, which correspond to the aorta, are analogous to the protein backbone in SCP. Naturally, the branches of the aorta are analogous to the side-chains of the protein. The deviation score between a pair of joint-sharing vessel parts is analogous to the covalent energy between bonded atoms. The overlap score between any pair of vessel parts plays exactly the same role as Van der Waals energy that avoids atomic overlaps. Therefore, when a set of n vessel parts is

given and the geometry of the "backbone" is fixed, the score function (1) can be reexpressed using the SCP terminology

$$\mathcal{E}(\mathbf{r}) = \sum_{i=1}^{n} \mathcal{E}_{\text{self}}(r_i) + \sum_{i<j}^{n} \mathcal{E}_{\text{pair}}(r_i, r_j), \tag{7}$$

where r_i denotes a rotamer at the ith residue; $\mathbf{r} = (r_1, \ldots, r_n)$; $\mathcal{E}_{\text{self}}(r_i)$ is the self-energy of the rotamer as well as its interaction energy with the fixed backbone; and $\mathcal{E}_{\text{pair}}(r_i, r_j)$ is the pairwise interaction energy between the rotamers for the ith and the jth residues. Each energy term is a sum of the vessel part-level score functions (3), (4), and (6).

There is, however, a key difference: in USIV, the library of favorable configurations cannot be constructed from nature. Therefore, the rotamer library for the USIV optimization problem needs to be built computationally to bear the desired properties.

4.2 Rotamer Library Generation

The rotamer library for the USIV optimization problem can be constructed computationally using Monte Carlo sampling.

For the set V_i of p vessel parts $v_1, \ldots, v_p$ that belong to the ith branch and the backbone, consider the score function (1) restricted to this branch, with the other branches fixed

$$\mathcal{E}_i(\Theta_{v_1}, \ldots, \Theta_{v_p}) = \mathcal{E}(\Theta_1, \ldots, \Theta_{|V|} | \Theta_v = \Theta_v^{(\text{ref})}, \\ \text{for } v \in V \text{ but } v \notin V_i). \tag{8}$$

In terms of SCP, this function is equivalent to the self-energy $\mathcal{E}_{\text{self}}(r_i)$, where $r_i = (\Theta_{v_1}, \ldots, \Theta_{v_p})$. It is desirable for the sampled configurations of the branch to have low values of the restricted score function (8), analogous to the rotamers in SCP.

We sample the branch configurations in the low-score regions of the configuration space in which the restricted score function (8) is defined, allowing some high-score configurations that may minimize the overall score function (1). To do this, we sample configurations from the Boltzman distribution for which the probability density of a configuration r_i given by

$$f(r_i) = e^{-\mathcal{E}_i(r_i)/T}/Z(T).$$

Here, T is "temperature," a parameter that determines the spread of the samples, and $Z(T)$ is the normalization constant that depends on the temperature. Sampling from an arbitrary probability distribution is performed using the acceptance-rejection (A-R) sampling, a simple Monte Carlo method. Fig. 3 illustrates this sampling strategy.

4.3 Solution Method via Integer Linear Programming

Once a rotamer library is generated, we can find the global minimum energy conformation of branches given the rotamer library using global optimization methods. This section follows the ILP formulation proposed by Kingsford et al. [26], originally developed for SCP.

The ILP formulation is easy to explain using graph theoretic terms. Each rotamer r_i at the ith residue is represented by a vertex u. We denote all possible rotamers at the ith residue by the vertex set V_i, deliberately duplicating the definition in Section 4.2. For each vertex u in V_i, the self-energy of the corresponding rotamer $\mathcal{E}_{uu} = \mathcal{E}_i(r_i) = \mathcal{E}_{\text{self}}(r_i)$
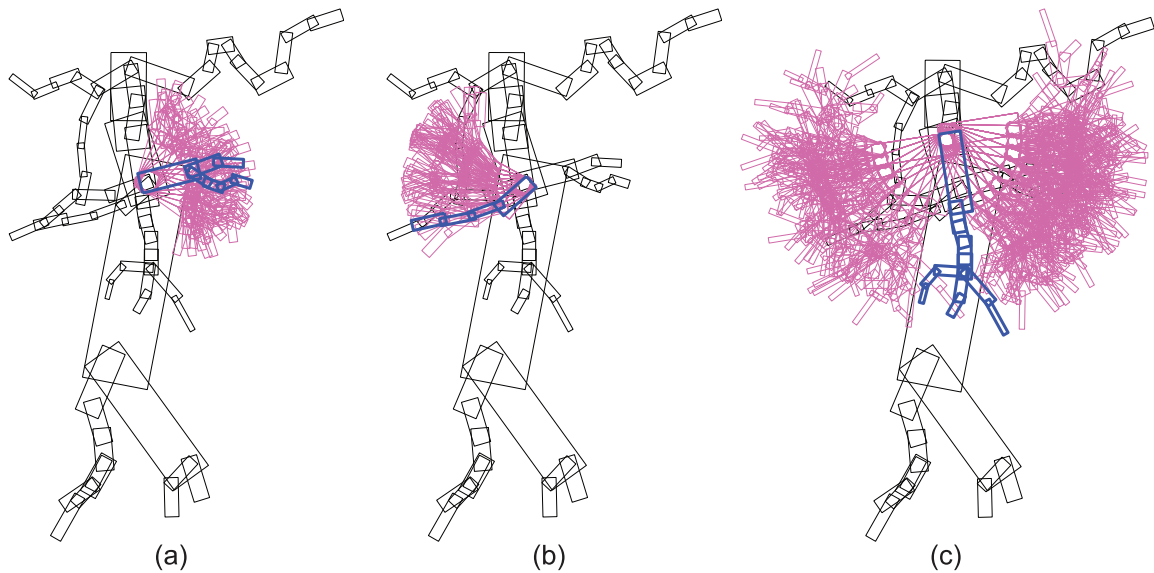
Fig. 3. An illustration of rotamer library generation. Thick blue boxes represent the reference configuration of the vessel parts corresponding to each branch. Thin purple boxes represent the sampled configurations. Black boxes are other vessel parts held fixed in the sampling process. (a) Right renal artery. (b) Left renal artery. (c) Superior mesenteric artery.

is given as a weight. For each pair of vertices $u \in V_i$ and $v \in V_j$, with $i \neq j$ and V_i, V_j disjoint, which correspond to rotamers r_i and r_j at different residues, an edge joins the two vertices with a weight of the pairwise energy $\mathcal{E}_{uv} = \mathcal{E}_{ij}(r_i, r_j) = \mathcal{E}_{\mathrm{pair}}(r_i, r_j)$, w h e r e $\mathcal{E}_{ij}(r_i, r_j) = \mathcal{E}(\Theta_1, \ldots, \Theta_{|V|}|\Theta_v = \Theta_v^{(\mathrm{ref})}$ for $v \in V \backslash (V_i \cup V_j))$. The resulting graph G is an undirected n-partite graph with vertex set $V = V_1 \cup \cdots \cup V_n$. The edge set of the graph G is denoted by E. Constructing a conformation of the side-chains corresponds to selecting one and only one vertex from each V_i, $i = 1, \ldots, n$. The remaining task is to pick the set of such vertices that corresponds to the global minimum energy conformation.

By introducing binary indicator variables, we can mathematically formulate the desired task. Let a binary variable x_{uu} indicate whether vertex (rotamer) u is selected: $x_{uu} = 1$ means that u is selected. Similarly, let x_{uv} be the binary indicator variable for edge (u, v). Then, the function

$$\mathcal{E}(\{x_{uu}\}_{u \in V}, \{x_{uv}\}_{(u,v) \in E}) = \sum_{u \in V} \mathcal{E}_{uu} x_{uu} + \sum_{(u,v) \in E} \mathcal{E}_{uv} x_{uv}$$

is equivalent to (7), provided that 1) only one vertex is chosen from each V_i, and 2) edge (u, v) is chosen if and only if vertices u and v are both selected. The first condition can be written in terms of the binary indicator variables

$$\sum_{u \in V_j} x_{uu} = 1, \quad j = 1, \ldots, n.$$

The second condition can also be expressed in terms of the binary indicator variables

$$\sum_{u \in V_j} x_{uv} = x_{vv}, \quad v \in V \backslash V_j, j = 1, \ldots, n.$$

If vertex v is not selected ($x_{vv} = 0$), then no edges adjacent to v can be chosen. If vertex v is chosen ($x_{vv} = 1$), only one edge adjacent to v can be chosen for each V_j. Then, the adjacent vertex in V_j must be chosen because of the

reciprocity; the above equation will still be satisfied if u and v are switched, and $x_{uv} = x_{vu}$ since the graph G is undirected.

Then, minimization of the function (7) is now a 0-1 integer linear program

$$\text{minimize} \sum_{u \in V} \mathcal{E}_{uu} x_{uu} + \sum_{(u,v) \in E} \mathcal{E}_{uv} x_{uv}$$
$$\text{subject to} \sum_{u \in V_j} x_{uu} = 1, \quad j = 1, \ldots, n$$
$$\sum_{u \in V_j} x_{uv} = x_{vv}, \quad v \in V \backslash V_j, j = 1, \ldots, n \qquad (9)$$
$$x_{uu}, x_{uv} \in \{0, 1\}.$$

Although this reformulation does not make the problem less complex (ILP is NP-hard [27]), there are at least two practical advantages of the ILP formulation over solving the original problem through SA. First, heuristic techniques for ILP, such as linear programming relaxation, can be applied to USIV. These techniques are mature enough to be employed in commercial, general-purpose ILP solvers, e.g., IBM ILOG CPLEX [28], and can be used to solve large-scale problems [29]. The second, big, advantage is parallelizability. Problem (7) requires precomputing $\mathcal{E}_{uu}$ and $\mathcal{E}_{uv}$ for a given rotamer library. This precomputation can be easily parallelized using general purpose graphics processing unit (GPGPU) techniques. In contrast, SA is less obvious to parallelize in general since it is an inherently sequential algorithm.

5 IMPLEMENTATION

5.1 Parallelizing Precomputation on GPU

We implemented the parallel precomputation of $\mathcal{E}_{uu}$ and $\mathcal{E}_{uv}$ using CUDA [30]. In CUDA, a program consists of a host program and one or more parallel kernels running on GPU cores. Each kernel executes a sequential program on a set of parallel threads, which are organized in a hierarchical fashion: a block is a set of threads and a grid is a set of

TABLE 1
Patient Scan (Left) and Algorithm (Right) Parameters

Parameter	Value	Category	Parameter	Value
Pixel grid	512 x 512	Subdivision	aspect ratio	10
In-plane field of view	299 – 447 mm	Total deviation weight	λ	4E-3
Section thickness	0.625 – 1.25 mm	Joint-sharing overlap weight	α	2E-4
No. slices per patient	508 – 693	Joint-sharing pairwise overlap linear region	a	0.2
			b	0.5

blocks. This hierarchy of programming threads matches up well with the memory hierarchy of GPUs. The threads inside each block can coordinate using a shared memory, and threads from different blocks in the same grid can communicate via a global memory. As the latency of accessing memory increases as traversing down the hierarchy, it is crucial to make threads exploit the limited shared memory space as much as possible. To this end, we parallelized the steps for computing self-energy and pairwise energy by assigning an instance of energy computation to each thread and then had multiple threads run in parallel. Since there is no dependency between such computation instances, there exists abundant parallelism that GPUs can exploit. The self-energy ($\mathcal{E}_{uu}$) of a rotamer is a sum of two components: 1) the energy between the vessel parts within a rotamer and 2) the energy between the vessel parts of a rotamer and those of the backbone. In computation of 1, each thread is mapped to a pair (x, y) of vessel parts in the rotamer and is indexed by (x, y). For 2, each thread is mapped to a pair made of a vessel part in a rotamer and that in the backbone. Each thread reads in appropriate rotamer information from the global memory of the GPU device and then calculates the overlap and deviation for the corresponding pair of vessel parts. To obtain the final self-energy value for the rotamer, we employed a parallel sum reduction technique, in which each thread stores the computation result in its dedicated area in the shared memory of the GPU. The routine to compute the pairwise energy ($\mathcal{E}_{uv}$) between two rotamers is parallelized in a similar manner: we let the block indexed by (A, B) handle the pairwise energy between rotamers A and B, and thread (x, y) inside this block computes the overlap and deviation values between vessel parts x in rotamer A and y in rotamer B.

5.2 Software Implementation

The algorithm was implemented with Matlab and C under Ubuntu Linux 10.10 environment. The GPU parallelization code was written using CUDA Toolkit 4.0. The ILP was solved using IBM ILOG CPLEX 12.2. The system used has a 2.66 GHz Intel Core i5 CPU with 8 GB memory. The GPU used was Nvidia GeForce GTX465 that has 352 computing cores (11 multiprocessors and 32 scalar processors per multiprocessor) and 1 GB of GDDR5 memory.

6 EVALUATION

We next present an evaluation of this method for the task of visualizing CTA scans of the aortic vessel tree.

6.1 Patient Data and Reference

We used the same data set as for the evaluation of the SA-based method in [1]. Briefly summarized, we retrospectively collected five patient scans, chosen to contain normal and typical anatomic variation, from our CTA archive. Table 1 provides the CT scan parameters. Under the approval of our Institutional Review Board, all the CT images were anonymized as required by HIPAA. We segmented the CTA images of the abdominal aortic vessel tree using the vascular modeling toolkit (VMTK) [31], based on [20]. The major aortic branches were labeled from the resulting segmentation by a radiologist who specialized in vascular imaging.

6.2 Generating Outputs for the Patient Data

Using the algorithm described in Section 4, we generated uncluttered single-image visualizations for each of the five cases. For each backbone site, or branch, 50, 100, 150, 200, and 400 rotamers were, respectively, sampled in a manner described in Section 4.2. The parameters for the score function are shown in Table 1.

6.3 Using the Score to Determine Candidate Visualizations for Evaluation

Although the ILP formulation deterministically provides the global minimum energy conformation of branches given a rotamer library, the generation of the rotamer library using a Monte Carlo method makes the method essentially stochastic. To address its stochastic nature, we generated 100 rotamer libraries for each of the five patient scans and selected the best three and the worst three of the 100 outputs, in terms of the score function (1) of the outputs, resulting in a total of six variants for each of the five patient scans.

6.4 Quality Metrics

6.4.1 Overlap Metrics

The overlap among the branches in the reference configuration and the configuration produced were evaluated according to the following metrics.

Centerline intersection counts the intersection of the centerlines in each branch of the backbone with the centerlines of all other branches. This metric does not consider an overlap due to the diameter of the branches.

Overlap proportion measures the overlap due to the diameter of the branches. For every vessel part v in a given branch b_i of the backbone, and for every vessel part w of the other branch b_j of the backbone, we compute the proportion A_{vw}/A_v of the area of intersection to that of the corresponding bounding box, where A_{vw} and A_v are as defined in Section 3.2. The average of the proportions of intersection, denoted r_{ij}, is roughly a measure of the proportion of b_i

overlapped by b_j owing to their widths. Finally, we take the average of r_{ij} over all b_js, weighted by the number of vessel parts in b_j, and the maximum of r_{ij} over all b_js and use this as the overlap proportion metrics of b_i in the presence of all the other branches. Since average ratio r_{ij} overestimates the proportion of the area of b_i overlapped by b_j, this metric may exceed unity.

6.4.2 Deviation Metrics

The deviation of the configuration our method produces from the reference configuration was evaluated branch-by-branch by the following metrics.

Branch distortion measures the overall angular distortion within the branch for every joint-sharing pair of vessel parts (v, w). We compute the deviation angle of the relative orientations, given by $|\theta_{vw} - \theta_{vw}^{(\text{ref})}|$, where $\theta_{vw} = \Theta_v - \Theta_w$ and $\theta_{vw}^{(\text{ref})} = \Theta_v^{(\text{ref})} - \Theta_w^{(\text{ref})}$ are the relative angles between the pair of parts. For each branch of the backbone, we take the average of the deviation angles of the joint-sharing part pairs. The deviation angle of the part corresponding to the root of the branch is not included in the averaging.

Branch rotation measures the rotation of the whole branch with respect to the backbone. We use the deviation angle of the part corresponding to the root of a branch.

6.5 Statistical Methods

We compared the SCP-based USIV method with the SA-based one in terms of execution time and the quantitative metrics presented above. Execution time was compared statistically using Wilcoxon's signed rank test [32]. The quantitative metrics were tested using the test of noninferiority, under the null hypothesis that the SCP-based method is more than 15 percent worse than the SA-based one.[1]

7 RESULTS

7.1 Visual Display of Outputs

Fig. 4 provides a visual display of the outputs of the proposed SCP-based method for the library size of 150 rotamers per branch. Why this library size is chosen is discussed in Section 7.3. All of the 30 visualizations are presented by case and by variant, sorted from the lowest scoring (best) to the highest scoring (worst).

7.2 Quality Metrics

The overlap metrics, reported in Supplemental Table S1, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TVCG.2012.25, agreed with visual inspection of Fig. 4: no crossing of the branches was found. The SCP-based USIV method completely removed the intersections among the centerlines for all the variants in each of the five cases. It also significantly reduced the average and the maximum overlap proportion metrics. Both the quantitative results and the visual outcomes are almost identical to those of the SA-based method (See [1, Table 4 and Fig. 5]).

Comparison of the deviation metrics to those of the SA-based method suggests that their performance in terms of

these metrics is very similar. Supplemental Table S2, available online, presents a case-by-case comparison, with regard to the branch distortion and the branch rotation metrics, averaged over the branches, of the SA-based method and the SCP-based method. Statistical test results support the observation. In terms of these metrics, the SCP-based method was not inferior to the SA-based method (branch distortion: $p = .0114$, test of noninferiority with 15 percent tolerance; branch rotation: $p = .0311$, the same test of noninferiority).

7.3 Impact of ILP Formulation and GPU Parallelization

The combination of ILP formulation and GPU parallelization achieves significant reduction in computation time for USIV. Table 2 summarizes the total computation time, which is the sum of the library generation time, the energy precomputation time for the self-energy and pairwise energy terms, and the ILP solving time, for each of the five cases and for library sizes of 50, 100, 150, 200, and 400 rotamers per branch. As a benchmark, the total computation time of the SA-based method is also presented. The total computation time increases with the library size. Library sizes smaller than 150 rotamers per branch did not completely resolve the overlap and the size of 400 exhibits a severe variance and may take longer than the SA-based method. Therefore, 150 appears to be a good tradeoff. The SCP-based method with this library size was significantly faster than the SA-based one ($p < .000001$, Wilcoxon signed rank test). At the library size of 150, the SCP-based method was 5.85 to 36.5 times faster than the SA-based method, roughly an order of magnitude improvement.

To understand the contribution of each step in computation, we plot in Fig. 5a the cumulative computation time of each of library generation, energy precomputation, and ILP solver, versus library size, for case C that has a median rank among the five in the total computation time. Rotamer library generation takes the smallest portion while the time required to solve the ILP rapidly increases. Similar trends were observed for the remaining cases.

The energy precomputation step demonstrates the power of GPU parallelization. In an early attempt, in which it was implemented sequentially, precomputation took longer and the majority of the total computation time. This was not unexpected since the (sequential) precomputation time is quadratic in the number of rotamers per branch. The GPU parallelization overcomes this drawback. To further understand the merit of parallelization, we conducted an experiment that compares the precomputation time of the sequential and parallel implementations by varying the library size from 50 to 1,000, incremented by 50. The result is shown in Fig. 5b for case C. Due to the long computation time of the sequential implementation, only a single library is generated for each size. Despite this limitation, it seems clear that it is the GPU parallelization that drives the large improvement in total computation time. As the number of rotamers per residue increases, the performance gap becomes wider, and, for the size of 1,000, the parallel implementation outperformed the serial one by approximately 120 times (43,158 seconds versus 354 seconds). Similar phenomena were observed for the other cases.

CPLEX solved the ILP exactly for all the generated rotamer libraries. It found the optimal solutions with high

---

1. A 15 percent margin in the inferiority test at 5 percent significance level implies that there is a 5 percent chance that SCP is up to 15 percent inferior to SA.
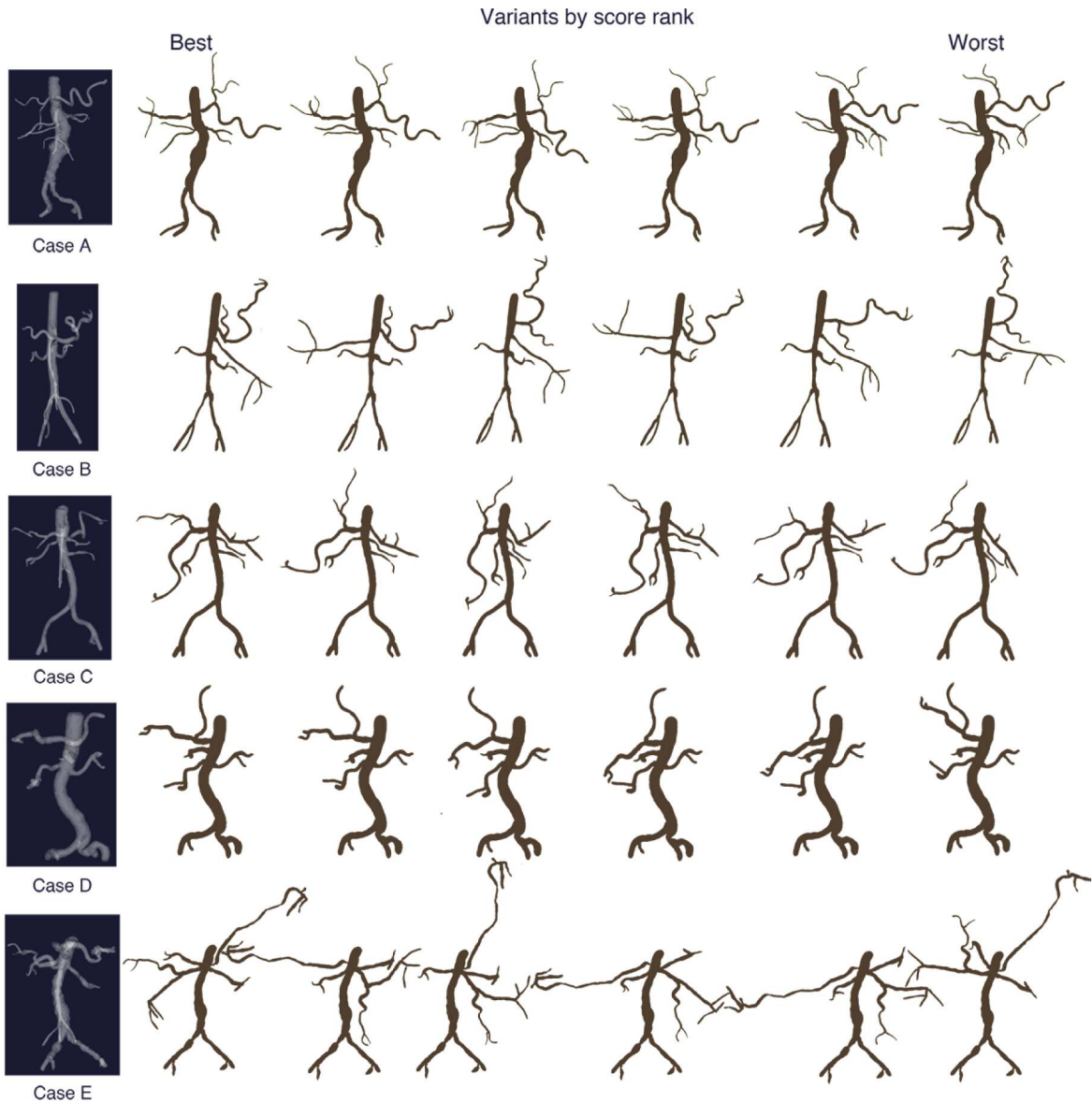
Fig. 4. Visual display of the outputs of the SCP-based method with the library size 150. Each row of the array of visualizations begins with the anteroposterior projection of the abdominal aortic vessel tree reconstructed from the segmentation and the surface extraction. The USIV outputs of the corresponding vessel tree are presented, arranged with the score ranking, from the best to the worst.

probabilities using the LP relaxation, i.e., which replaces the binary constraint $x_{uu}, x_{uv} \in \{0,1\}$ in (9) with the interval constraint $0 \le x_{uu}, x_{uv} \le 1$; see Table 3. This result agrees with that of [26], which reports 60 to 100 percent success rates of the LP relaxation. Even when the LP relaxation did not find the optimum in the first place, CPLEX found it through subsequence branch-and-bound steps within 50 percent of the time needed to solve the first relaxed LP. As can be seen in Table 3, success of the LP relaxation did not largely depend on the library size; it was the size of the LP that mostly determines the solving time. Although the ILP solving time can increase dramatically as the size grows, at the library size of choice (150) it was not a severe burden.

8 DISCUSSION

8.1 Quantitative Metrics

While it would require a major effort to prove that the SCP-based method is equivalent or perhaps superior to established visualization methods, our results indicate that it does provide results that are similar to those we obtained using our SA-based method, which clinicians did find acceptable [34]. In this reader study, three trained radiologists who were familiar with the 3D anatomy of the abdominal aorta independently rated distortion of the branches in the 30 visualizations generated in a similar manner from the same five data sets, in order to see if the visualizations generated by the method are acceptable to clinicians. A regression analysis showed that both the branch rotation and the branch distortion metrics were significant predictors of a reader's subjective distortion rating with positive correlations ($p < .0001$ for both factors, using the Generalized Estimating Equations regression model). The monotone increasing trend between the subjective distortion rating and each of the deviation metrics was found to be significant ($p < .0001$ for both metrics, Jonckheere-Terpstra test). In other words, branches with a lower distortion rating had lower branch rotation
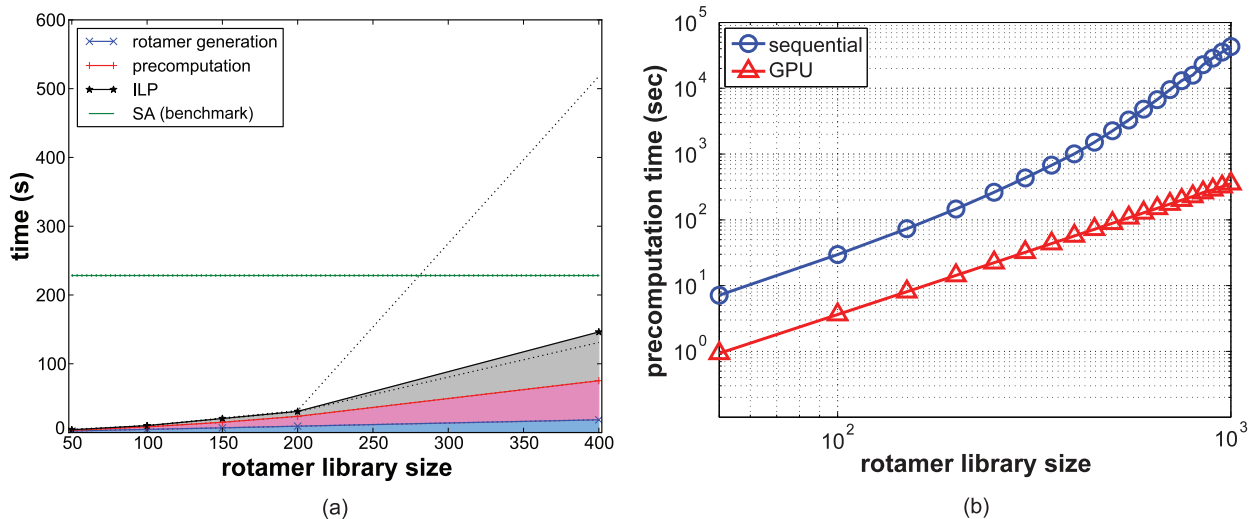
Fig. 5. Computation time by library size (case C). (a) Cumulative computation time of each step. Solid lines represent the median time, with the band made of dotted lines represents 25 and 75 percent quantiles. As a benchmark, the corresponding quantities for the SA-based method are plotted as horizontal lines. (b) Comparison of serial versus GPU-parallelized precomputation time.

TABLE 2
Total Computation Time* of the SCP-Based Method Compared to Those of the SA-Based One

Case	SA	SCP rotamer library size				
		50	100	150	200	400
A	307.5	3.157	10.64	23.05	48.12	214.1
	(306.0,333.8)	(2.963,3.605)	(9.479,12.27)	(20.57,26.02)	(38.64,65.08)	(199.2,259.4)
B	228.4	7.404	19.67	13.24	22.59	548.7
	(228.1,229.1)	(5.601,9.316)	(17.71, 23.17)	(12.85, 13.69)	(21.89, 23.45)	(124.1, 656.4)
C	228.3	4.057	10.08	20.44	30.73	146.2
	(227.4, 229.4)	(3.831, 4.823)	(9.870, 10.26)	(20.10, 21.17)	(30.17, 32.02)	(131.0, 517.9)
D	176.6	8.541	18.33	30.16	41.98	127.6
	(176.2, 177.0)	(7.864, 9.366)	(17.21, 19.58)	(28.70, 31.80)	(40.52, 44.87)	(122.9, 135.5)
E	669.5	2.865	8.517	18.32	26.45	141.0
	(664.8, 700.5)	(2.736, 3.217)	(7.986, 8.833)	(17.06, 20.00)	(24.60, 29.45)	(112.8, 165.0)

* Median (25% quantile, 75% quantile), in seconds.

and lower branch distortion metrics. Since the outputs of the SCP-based method behave similarly to those of the SA-based one, we expect that clinicians would perceive them similarly.

8.2 Choice of the Weight of the Total Deviation

The question of how sensitive the results of optimization are with respect to the choice of a particular λ, the weight of the total deviation, is discussed in [1]. We observed essentially the same behavior in those of the SCP-based

TABLE 3
Success Rate (Percent) of the LP Relaxation

Case	Rotamer library size				
	50	100	150	200	400
A	74	62	68	68	90
B	100	99	100	100	100
C	99	100	100	99	100
D	99	98	97	99	97
E	100	96	98	94	95

method: moderate change (-20 percent to +20 percent) from the value in Table 1 did not change the results very much; most importantly, they did not introduce new intersections. However, an order of magnitude increase did not completely remove overlaps. It also increased the ILP solving time by 1.4 to 3.9 times at the chosen library size. This is expected since the larger the weight is given to the total deviation, the harder the optimization problem to solve.

8.3 Potential Impact of ILP to Other Visualization Problems

Our formulation of the visualization problem as an integer program and use of parallel precomputation may have potential value for other applications in visualization, e.g., label placement [27], [35], [36], and drawing of metro maps [37]. In particular, [27], [37] utilize integer programming similar to ours. In principle, if the problem is formulated as a mathematical program whose objective function is a linear combination of 0-1 variables that partition such that each partition is constrained to sum to 1, then the problem can benefit from our approach. This is precisely the case for [27], where each of the K labels to be placed takes only one of the N_k potential positions, $k = 1, \ldots, K$. The weights of the

linear combination $W_{i,k}$, $i = 1, \ldots, N_k$, $k = 1, \ldots, K$ can be precomputed in parallel using GPU. Then, the resulting ILP can be solved using CPLEX, without resorting to various heuristics discussed in [27]. The implication of this recipe is twofold: 1) N and/or K can be large as long as CPLEX can handle, thereby enabling more labels with finer granularity; 2) $W_{i,k}$ can be rapidly recomputed if the underlying scene changes, potentially enabling dynamic labeling. In [37], which uses a fixed weighting, the formulation could be improved to assign edge-specific weights that encode preference in bending and position of the edge, adaptive to underlying geography on which the metro map can be overlaid (This potential extension is not discussed in [37]).

8.4 Future Work

Although we chose to accurately segment the geometry of the aorta before applying the method, in principle our approach of simplifying the geometry using bounding boxes can be used on models generated by isosurface extraction (e.g., Marching Cubes). Since requiring a segmentation may limit the future practical applicability of the technique, further efforts in this direction will be fruitful.

Along this line of thought, a conservative approximation of the overlap between two vessel segments can be made in pixel-precise level on GPUs due to their superior rasterization performance. In fact, our initial efforts of utilizing GPUs were devoted to pixel-precise overlap area calculation. However, it turned out that retrieving the number of overlapping pixels from GPU to CPU is a performance bottleneck since the overlap area has to be calculated and transferred thousands of times. If this bottleneck could be lifted, this might improve the overall distortion since it offers more space to actually place the vessels than using bounding boxes.

Other avenues for parallelization can be explored as well. First, rotamer library generation may be parallelizable on GPUs, although the profiling of total computation time shown in Fig. 5a suggests the overall speedup would be marginal. Second, since solving the ILP has become the limiting factor in achieving an interactive rate of rendering, parallelizing this part will be more fruitful. The current implementation provides a single view within a minute. Although it appears that producing multiple views violates our original motivation of single-image visualization, if achieved, an interactive rate for a single view should generate multiple views in tolerable amount of time. To this end, specialized solution methods for solving the particular ILP (9), e.g., [38], [39], instead of general ILP solvers, may well be exploited.

Another interesting direction for future research is to investigate if our approach could be applicable to more complex vascular trees or to vascular networks with loops, such as the liver vasculature or the cerebral circulation. In particular, the latter may unavoidably lead to overlaps due to topological constraints. Currently, our method is limited to not too large trees for which the topology allows planar embedding. However, if the topology allows successive approximation with trees, our SCP-based method would serve a good progressive visualization method since the approach of placing the backbone first and residues later plays an important role in the success of the algorithm for the aortic tree.

A limitation of our study is that we only investigated producing results for the canonical reference direction for the aortic vasculature (anteroposterior). While our previous paper reported minimal sensitivity to small deviations (<30 degrees) from this reference direction [1], robustness for other directions, if desired, remains a subject for future work.

9 CONCLUSION

This paper has introduced a novel solution method for uncluttered single-image visualization, inspired by the connection between USIV and protein side-chain placement, and illustrated its use for the visualization of the aortic vessel tree as imaged by CTA or MRA. The proposed method is built within the framework for USIV introduced in [1]: geometric simplification of the vessel tree and use of articulated object models with a minimal anatomical assumption. This method eliminates misleading false intersections among the aortic vessel branches by formulating the global optimization problem for USIV as an integer linear programming, borrowed from the SCP literature. The ILP formulation provides an ample room for parallelism that can be exploited using commodity devices such as GPUs, and allows use of industrial-strength solvers. This combination of techniques has yielded a significant improvement in the total computation time of the algorithm, with equivalent output visualizations in terms of deviation metrics compared to the method presented in [1].

It is our impression that integer programming has not been much considered in visualization and computer graphics applications. This may be because of the belief that the combinatorial optimization problem is not tractable in practical computational settings [29]. With the advance of off-the-shelf ILP solvers and ubiquity of commodity-level parallelism, many integer programming problems can now be solved practically. We hope that our results are encouraging enough for the community to take another look at the use of mathematical programming for applications in visualization.

REFERENCES

[1] J. Won, J. Rosenberg, G.D. Rubin, and S. Napel, "Uncluttered Single-Image Visualization of the Abdominal Aortic Vessel Tree: Method and Evaluation," *Medical Physics*, vol. 36, no. 11, pp. 5245-5260, Nov. 2009.

[2] M. Tory, A.E. Kirkpatrick, M.S. Atkins, and T. Möller, "Visualization Task Performance with 2D, 3D, and Combination Displays," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 1, pp. 2-13, Jan./Feb. 2006.

[3] A. Kanitsar, R. Wegenkittl, D. Fleischmann, and M.E. Gröller, "Advanced Curved Planar Reformation: Flattening of Vascular Structures," *Proc. 14th IEEE Conf. Visualization*, pp. 43-50, Oct. 2003.

[4] R. Raman, S. Napel, C.F. Beaulieu, E.S. Bain, R.B. Jeffrey Jr., and G.D. Rubin, "Automated Generation of Curved Planar Reformations from Volume Data," *Radiology*, vol. 223, no. 1, pp. 275-80, 2002.

[5] X. Lan and D.P. Huttenlocher, "Beyond Trees: Common-factor Models for 2D Human Pose Recovery," *ICCV '05: Proc. 10th IEEE Int'l Conf. Computer Vision,* vol. 1, pp. 470-477, 2005.

[6] X. Ren, A. Berg, and J. Malik, "Recovering Human Body Configurations Using Pairwise Constraints between Parts," *Proc. 10th IEEE Int'l Conf. Computer Vision,* vol. 1, pp. 824-831, Oct. 2005.

[7] A. Gupta, A. Mittal, and L. Davis, "Constraint Integration for Efficient Multiview Pose Estimation with Self-Occlusions," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 3, pp. 493-506, Mar. 2008.

[8] O. Karpenko, W. Li, N. Mitra, and M. Agrawala, "Exploded View Diagrams of Mathematical Surfaces," *IEEE Trans. Visualization and Computer Graphics,* vol. 16, no. 6, pp. 1311-1318, Nov./Dec. 2010.

[9] W. Li, M. Agrawala, B. Curless, and D. Salesin, "Automated Generation of Interactive 3D Exploded View Diagrams," *Proc. ACM SIGGRAPH,* pp. 1-7, 2008.

[10] M. Agrawala, D. Phan, J. Heiser, J. Haymaker, J. Klingner, P. Hanrahan, and B. Tversky, "Designing Effective Step-by-Step Assembly Instructions," *ACM Trans. Graphics,* vol. 22, no. 3, pp. 828-837, 2003.

[11] C. Ware and P. Mitchell, "Visualizing Graphs in Three Dimensions," *ACM Trans. Applied Perception,* vol. 5, no. 1, pp. 1-15, 2008.

[12] C. Ware and P. Mitchell, "Reevaluating Stereo and Motion Cues for Visualizing Graphs in Three Dimensions," *Proc. Second Symp. Applied Perception in Graphics and Visualization,* pp. 51-58, 2005.

[13] A.V. Bartrolí, R. Wegenkittl, A. König, and E. Gröller, "Nonlinear Virtual Colon Unfolding," *Proc. IEEE Conf. Visualization,* pp. 411-420, 2001.

[14] L. Zhu, S. Haker, and A. Tannenbaum, "Flattening Maps for the Visualization of Multibranched Vessels," *IEEE Trans. Medical Imaging,* vol. 24, no. 2, pp. 191-198, Feb. 2005.

[15] M. Termeer, J.O. Bescós, M. Breeuwer, A. Vilanova, F. Gerritsen, and M.E. Gröller, "CoViCAD: Comprehensive Visualization of Coronary Artery Disease," *IEEE Trans. Visualization and Computer Graphics,* vol. 13, no. 6, pp. 1632-1639, Nov./Dec. 2007.

[16] T. Ropinski, S. Hermann, R. Reich, M. Schafers, and K. Hinrichs, "Multimodal Vessel Visualization of Mouse Aorta PET/CT Scans," *IEEE Trans. Visualization and Computer Graphics,* vol. 15, no. 6, pp. 1515-1522, Nov./Dec. 2009.

[17] S. Oeltze and B. Preim, "Visualization of Vasculature with Convolution Surfaces: Method, Validation and Evaluation," *IEEE Trans. Medical Imaging,* vol. 24, no. 4, pp. 540-548, Apr. 2005.

[18] H. Hahn, B. Preim, D. Selle, and H. Peitgen, "Visualization and Interaction Techniques for the Exploration of Vascular Structures," *Proc. 11th IEEE Conf. Visualization,* pp. 395-402, 2001.

[19] T. Deschamps, "Extraction de Courbes et Sufaces par méthodes de Chemins Minimaux et Ensembles de Niveaux. Applications en Imagerie Medicale 3D," PhD dissertation, L'université de Paris-Dauphine, 2001.

[20] L. Antiga, "Patient-Specific Modeling of Geometry and Blood-Flow in Large Arteries," PhD dissertation, Politecnico di Milano, 2003.

[21] M. Levitt and A. Warshel, "Computer Simulation of Protein Folding," *Nature,* vol. 253, no. 5494, pp. 694-698, 1975.

[22] C. Yanover and Y. Weiss, "Approximate Inference and Protein-Folding," *Advances in Neural Information Processing Systems,* S. Becker, S. Thrun, and K. Obermayer, eds., vol. 15, pp. 1457-1464, 2002.

[23] D.B. Gordon, S.A. Marshall, and S.L. Mayo, "Energy Functions for Protein Design," *Current Opinion in Structural Biology,* vol. 9, no. 4, pp. 509-513, 1999.

[24] R.L. Dunbrack Jr. and M. Karplus, "Conformational Analysis of the Backbone-Dependent Rotamer Preferences of Protein Side-chains," *Nature Structural Biology,* vol. 1, no. 5, pp. 334-340, 1994.

[25] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne, "The Protein Data Bank," *Nucleic Acids Research,* vol. 28, no. 1, pp. 235-242, 2000.

[26] C.L. Kingsford, B. Chazelle, and M. Singh, "Solving and Analyzing Side-Chain Positioning Problems Using Linear and Integer Programming," *Bioinformatics,* vol. 21, no. 7, pp. 1028-1036, 2005.

[27] J. Christensen, J. Marks, and S. Shieber, "An Empirical Study of Algorithms for Point-Feature Label Placement," *ACM Trans. Graphics,* vol. 14, no. 3, pp. 203-232, July 1995.

[28] IBM, "IBM ILOG CPLEX Optimizer," 2011.

[29] D. Bertsimas and R. Shioda, "Classification and Regression via Integer Optimization," *Operations Research,* vol. 55, no. 2, pp. 252-271, Mar. 2007.

[30] J. Sanders and E. Kandrot, *CUDA by Example: An Introduction to General-Purpose GPU Programming.* Addison-Wesley, 2010.

[31] L. Antiga, "VMTK - Vascular Modeling Toolkit," 2006.

[32] F. Wilcoxon, "Individual Comparisons by Ranking Methods," *Biometrics Bull.,* vol. 1, no. 6, pp. 80-83, 1945.

[33] S. Wellek, *Testing Statistical Hypotheses of Equivalence and Non-inferiority.* Chapman & Hall, 2010.

[34] J. Won, "Uncluttered Single-Image Visualization of the Abdominal Aortic Vessel Tree Using Global Optimization Methods," PhD dissertation, Stanford Univ., Jan. 2009.

[35] M.A. Bekos, M. Kaufmann, A. Symvonis, and A. Wolff, "Boundary Labeling: Models and Efficient Algorithms for Rectangular Maps," *Proc. Int'l Symp. Graph Drawing,* J. Pach, ed., pp. 49-59, 2005.

[36] T. Stein and X. De'coret, "Dynamic Label Placement for Improved Interactive Exploration," *Proc. Sixth Int'l Symp. Non-Photorealistic Animation and Rendering,* pp. 15-21, 2008.

[37] M. Nollenburg and A. Wolff, "Drawing and Labeling High-Quality Metro Maps by Mixed-Integer Programming," *IEEE Trans. Visualization and Computer Graphics,* vol. 17, no. 5, pp. 626-641, May 2011.

[38] E.-J. Hong and T. Lozano-Pérez, "Protein Side-Chain Placement through Map Estimation and Problem-Size Reduction," *Proc. Workshop Algorithms in Bioinformatics,* pp. 219-230, 2006.

[39] S. Canzar, N.C. Toussaint, and G.W. Klau, "An Exact Algorithm for Side-Chain Placement in Protein Design," *Optimization Letters,* vol. 5, pp. 393-406, Mar. 2011.

Joong-Ho Won (S'02-M'09) received the BS degree in electrical engineering from Seoul National University, Korea, in 1998 and the MS and the PhD degrees in electrical engineering from Stanford University, CA, in 2003 and 2009, respectively. He also received the MS degree in statistics from Stanford University. Dr. Won was a research scientist at the VA Cooperative Studies Program, Mountain View, CA from 2011 to 2012. Currently, he is an assistant professor in the School of Industrial Management Engineering, Korea University, Seoul, Korea. His research interests are large-scale data analysis and visualization, and computational methods in medical informatics and bioinformatics. He is a member of the IEEE.

Yongkweon Jeon received the BS degree in electrical engineering from Korea University, Seoul, in 2009. Currently, he is a researcher at the Department of Electrical and Computer Engineering at Seoul National University and is working toward the MS degree in electrical engineering at Korea University. His research interests include clustering and parallel programming.

Jarrett K. Rosenberg received the PhD degree from UC Berkeley. He is a research scientist and biostatistician in the Stanford Radiology Department.

Sungroh Yoon (S'99-M'06-SM'11) received the BS degree in electrical engineering from Seoul National University, Korea, in 1996, and the MS and PhD degrees in electrical engineering from Stanford University, CA, in 2002 and 2006, respectively. From 2006 to 2007, he was with Intel Corporation, Santa Clara, CA. Previously, he held research positions with Stanford University, CA, and Synopsys, Inc., Mountain View, CA. Dr. Yoon was an assistant professor with the School of Electrical Engineering, Korea University from 2007 to 2012. Currently, he is an assistant professor with the Department of Electrical and Computer Engineering, Seoul National University, Korea. His research interests include bioinformatics and computational biology. He is a senior member of the IEEE.

Geoffrey D. Rubin received the BS degree with honors from the California Institute of Technology in 1982, the MD degree from the University of California at San Diego in 1987, and completed Radiology Residency and Body Imaging Fellowship Training at Stanford University in 1993. He is the George Barth Geller chair for Cardiovascular Research, the chairman of the Department of Radiology at the Duke University School of Medicine, and radiologist-in-chief at Duke University Hospital. Following this fellowship, he joined the faculty of the Stanford University School of Medicine in 1993, where he was granted tenure in 2000, promoted to professor in 2005, and served as the section chief of cardiovascular imaging until August 2010, when he moved to Duke University. He pioneered the use of spiral CT and multidetector-row CT for imaging the cardiovascular system and has personally performed and interpreted over 10,000 CT angiograms since 1991. He is the author of more than 165 peer-reviewed manuscripts and more than 50 review articles and book chapters.

Sandy Napel received the BS degree from the State University of New York at Stony Brook in 1974. He received the MS degree in 1976 followed by the PhD degree in engineering in 1981, both at Stanford University. He became an assistant professor in the Department of Radiology at UCSF in 1981, and was a founder of Imatron, Inc., developer of electron beam CT, where he served as VP of engineering. He joined the Department of Radiology at Stanford in 1991, where he is presently professor, with courtesy appointments in electrical engineering and medicine (biomedical informatics research), and where he codirects information science in imaging at Stanford (ISIS). He holds 14 patents, has authored or coauthored more than 100 peer-reviewed publications, eight book chapters, and more than 275 presentations at National and International Meetings. He has made contributions to the development of CT angiography, cardiac CT, visualization and quantitation of volumetric medical imaging data, computer-aided detection of lesions from medical imaging data, and is more recently interested in correlating image features to molecular biology.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.