Vision, Modeling, and Visualization (2014)
J. Bender, A. Kuijper, T. von Landesberger, and P. Urban (Eds.)

Pattern Search in Flows based on Similarity of Stream Line Segments *Additional Material*

Z. Wang, J. Martinez Esturo, H.-P. Seidel, and T. Weinkauf

Max Planck Institute for Informatics, Saarbrücken, Germany

Abstract

As a supplement of the paper, we give further details and analysis of the globally consistent stream line segmentation algorithm. Moreover, a more detailed description of the intrinsic similarity measure for comparing the resulting segments is also provided.

Notation. We make use of the following formal concepts: let $\mathbf{v}(\mathbf{x})$ denote steady differentiable vector fields with associated over two ($d = 2$) and three-dimensional ($d = 3$) flow domains $\mathcal{D} \subset \mathbb{R}^d$ with $\mathbf{x} \in \mathcal{D}$. Parametric *stream lines* $\mathbf{c}(t) = \mathbf{x}_0 + \int_0^t \mathbf{v}(\mathbf{c}(u))\,\mathrm{d}u$ are curves defined through integration along $\mathbf{v}$ starting from a seed point $\mathbf{x}_0$ for an integration time t. We partition stream lines $\mathbf{c}$ into disjoint *stream line segments* $\mathbf{s}_i(t)$ by splitting $\mathbf{c}$ at integration times t_i such that the points of $\mathbf{c}$ and $\mathbf{s}_i$ coincide for $t \in [t_i, t_{i+1}]$. For a stream line integrated from t_0 to t_n, a segmentation is defined by the sequence $[t_0, \ldots t_i, \ldots t_n]$ of segment boundaries t_i. We denote the length of $\mathbf{s}_i$ by l_i.

Figure 1: *Segment Merge Criteria. Pre-merge segment boundaries are colored (●), and two different average segment orientations are colored (●) and (●). (a) A pair of segments is mergeable if they both have similar average orientations. (b) A triplet of segments is mergeable if the center segment (●) has a low average total curvature compared to its neighboring segments, which have similar average orientations.*

1. Globally Consistent Segmentation of Stream Lines

1.1. Curve Curvatures.

In this work, we demonstrate that all curve segmentation requirements can be addressed by considering *intrinsic* curve properties only. In particular, the requirements can be expressed only in terms of curve *curvature*: feature locations along a curve are identified with high curvature values, whereas low curvature points along the curve are possible segmentation points. Also, segmentation consistency can be achieved using a curvature-based normalization.

Let $\dot{\mathbf{c}}(t) := \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{c} = \mathbf{v}$ and $\ddot{\mathbf{c}}(t) := \frac{\mathrm{d}^2}{\mathrm{d}t^2}\mathbf{c} = (\nabla\dot{\mathbf{c}})\,\dot{\mathbf{c}}$ denote the first two stream line derivatives in terms of vector field quantities (cf. [WT02]). Then stream line curvatures $\kappa_d(t)$ are given in two and three dimensions d by the well-known expressions

$$\kappa_2 = \frac{\det([\dot{\mathbf{c}}, \ddot{\mathbf{c}}])}{||\dot{\mathbf{c}}||^3} \qquad \kappa_3 = \frac{||\dot{\mathbf{c}} \times \ddot{\mathbf{c}}||}{||\dot{\mathbf{c}}||^3}\,. \tag{1}$$

The signed curvature κ_2 defines the geometry of 2D curves up to rigid transformations (see, e.g., [dC76]), which allows us to use κ_2 to represent their intrinsic geometry as a basis for segmentation. Although unsigned curvature κ_3 does not fully define 3D curves up to rigid transformations, we will show that for segmentation it is sufficient to only consider κ_3 for 3D curves. By only relying on curvature estimates, our segmentation scheme supports 2D and 3D curves in a unified way. Stream line segmentation proceeds in two phase, curvature-based splitting and subsequent segment merging, and we continue to describe both in more detail.

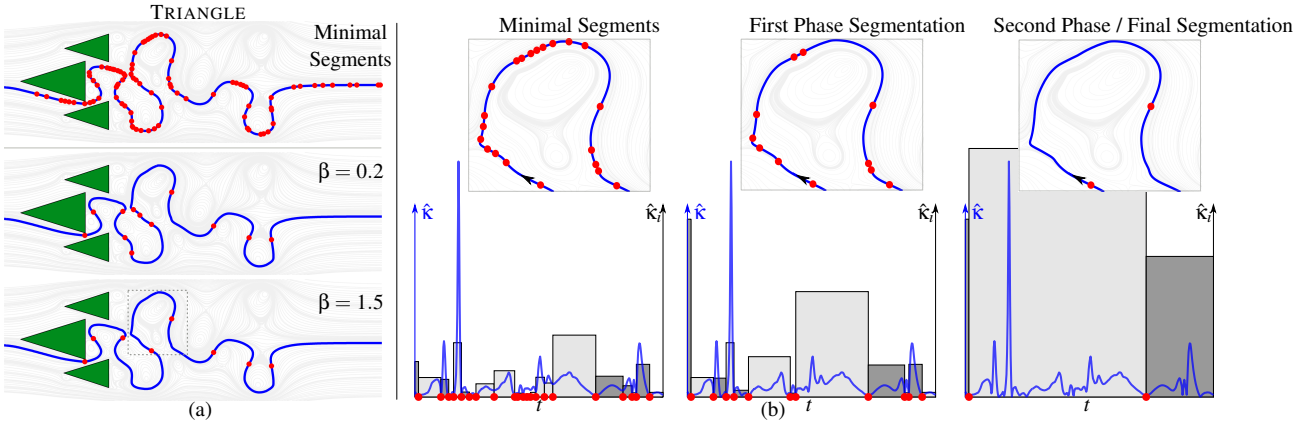

Figure 2: *Segmentation Scheme. (a) Starting from minimal segments (top) our segmentation scheme applies two phases of iterative segment merging. Shown are two results for different* β *parameters (bottom,* $\alpha = \pi/2$ *in both cases). (b) For the three segmentations of the* $\beta = 1.5$ *computation, the absolute curvatures* $\hat{\kappa}$ *(●) and total discrete segment curvatures* $\hat{\kappa}_i$ *(box height) together with average 2D segment orientation (positive ●, negative ●) of a cutout region are shown. In the first phase, neighboring segments of close average orientation are merged. In the second phase, compatible segments of low curvature are combined if its two neighbors are compatible w.r.t. average orientation.*

1.2. Segment Splitting.

Both curvature estimations κ_2 and κ_3 differ in their signedness. Hence, we consider absolute local curvatures $\hat{\kappa}(t) = |\kappa_{2/3}|$ for a unified stream line segmentation scheme that is applicable for both two and three dimensional stream lines. Vector field features are usually coupled to high absolute stream line curvatures (see, e.g., [MJL*13]). Therefore, to obtain feature-preserving and distinct segmentations, points of absolute local curvature minima that bound these high curvature regions are candidates for possible segment boundaries. Along a stream line, let t_i denote the integration times corresponding to absolute local curvature minima, i.e., $\hat{\kappa}(t) > \hat{\kappa}(t_i)$ for $t \in [t_i - \varepsilon, t_i + \varepsilon]$. We call these segments bounded by consecutive absolute local curvature minima *minimal segments*. Minimal segments are the initial building blocks of the final segmentation and will not be split further to preserve the features of higher curvature they represent. Still, as the total curvature of the features that minimal segments represent can vary considerably, we merge minimal segments into segments of higher significance.

1.3. Segment Merging.

We merge neighboring segments based on two segment properties: total segment curvature and average segment orientation. Both properties are scale-invariant and are therefore comparable for segments of different extend. The *total segment curvature* $\hat{\kappa}_i$ is given by

$$\hat{\kappa}_i = \int_{t_i}^{t_{i+1}} \hat{\kappa}\, ||\dot{\mathbf{c}}||\, \mathrm{d}t \ . \tag{2}$$

Note that $\hat{\kappa}_i$ is invariant to rigid transformations of the segment as the intrinsic local curvatures are already invariant to rigid transformations. Additionally, total curvature is also invariant w.r.t. scaling and inversion, as the segment velocity $||\dot{\mathbf{c}}||$ scales linearly and local curvatures scale inversely proportional to the scaling factor. Hence, total segment curvatures are invariant w.r.t. similarity transformations. This property allows us to relate segments in a scale-invariant and intrinsic way based on total curvature. In general, the total segment curvature is high for feature regions of a curve. Low total segment curvature will indicate candidate segments for merging.

Along each segment, the orthonormal Frenet-Serret frames $(\mathbf{t}(t), \mathbf{n}(t), \mathbf{b}(t))$ are given by the tangent, normal, and bi-normal directions, respectively (cf. [dC76]). For stream lines, tangent directions are given by the normalized vector field, normals point to the center of the osculating circle, and $\mathbf{b} = \mathbf{t} \times \mathbf{n}$. We observe that along a minimal segment the variation of bi-normal directions is usually small, e.g., for 2D curves they only vary at segment boundaries. Therefore, we assign an *average orientation* $\bar{\mathbf{b}}_i = \frac{\tilde{\mathbf{b}}_i}{||\tilde{\mathbf{b}}_i||}$ with

$$\tilde{\mathbf{b}}_i = \frac{1}{l_i} \int_{t_i}^{t_{i+1}} \mathbf{b}\, ||\dot{\mathbf{c}}||\, \mathrm{d}t \ , \tag{3}$$

to each segment, which is based on averaged bi-normal directions. Note that average orientations are scale-invariant. In addition, the angle $\alpha_i = \measuredangle(\bar{\mathbf{b}}_i, \bar{\mathbf{b}}_{i+1})$ between consecutive segments is invariant w.r.t. rigid transformations and is therefore invariant w.r.t. similarity transformations. We observe that averaged orientations don't vary significantly along important curve feature regions, which is considered for segment merging.

Our algorithm for merging of segments consists in growing segments of low total curvature with neighboring segments, if they are merge-compatible. Compatibility is tested in two phases based on two criteria: first, two neighboring segments

s_i and s_{i+1} are mergeable if they have similar average orientations, i.e., if the angle α_i is smaller than a user-specified upper bound α. This allows segments of similar average orientation to be extended: for instance, two smaller arcs can be combined to a bigger arc element, which increases the feature distinction of the segmentation (see Figure 1 (a)). Second, if one segment s_i has a low total curvature, i.e., $\hat{\kappa}_i < \beta$ for a user-specified upper bound β, it is mergeable with both of its neighboring segments s_{i-1} and s_{i+1} if these two segments have similar average orientations measured by α. This criterion allows to remove less significant segments of low average curvature while the newly created segment still respects the feature distinction property due to the preservation of average orientation (see Figure 1 (b)). The merging algorithm iteratively processes segments based on a priority queue that is ordered by the total segment curvature such that segments of lowest curvature are processed first. In the first phase, the first compatibility criterion is evaluated and compatible segments are merged. Merged segments are removed from the priority queue and the new segment is inserted with updated properties. Note that for closed stream lines, i.e., curves with coinciding endpoints, we consider the first and last segment to neighbor each other. The first phase terminates if no more segments can be merged. The iteration is repeated for the second compatibility criterion to give the final segmentation.

We illustrate the different steps of our segmentation scheme in Figure 2. The initial minimal segments of a single stream line are shown in Figure 2 (a) together with two segmentation results for different β values. This parameter steers the coarseness of the segmentation, and segmentations are usually not sensitive w.r.t. small β variations. Note that curve orientation is either positive or negative for all 2D curve segmentations. Hence, it is sufficient to select $\alpha = \pi/2$ for this case. For a closeup region, Figure 2 (b) shows segmentations and relation to local absolute curvatures after each segmentation phase. The graphs show that neighboring segments of similar orientation are merged into segments of higher total curvature. In the second phase, triplets of compatible segments are merged. Figure 3 depicts the consistency of segmentations along multiple stream lines for two different flows. The results illustrate that our segmentation consistently computes similar segments for similar flow patterns.

2. Intrinsic Similarity of Stream Line Segments

Based on our consistent curve segmentation scheme, we propose a general scale-invariant method for intrinsic curve segment comparison. Our flow pattern search approach will use segment similarities computed this way for pattern retrieval. Again, we represent the segment geometry using intrinsic curve properties only. This way the computed similarities are invariant to rigid transformations. In addition, we are comparing the intrinsic properties at equal scales to obtain scale-invariant similarities. Note that we don't require nor assume that segments originate from a common stream line for comparison. This will make our pattern search applicable

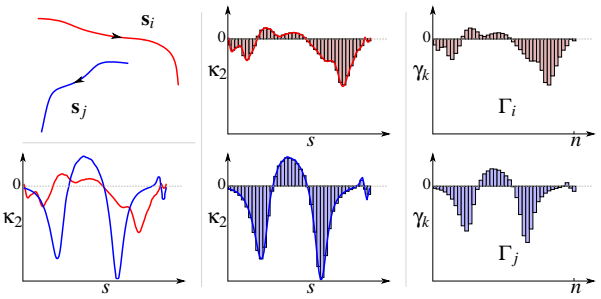

Figure 4: *Scale-invariant Intrinsic Profile Discretization. The signed curvatures (bottom left) of the two segments* $\mathbf{s}_i$ *and* $\mathbf{s}_j$ *of different lengths* $l_j < l_i$, $l_i = 1$ *are discretized into n bins of average property values (middle). Length normalization of the bin values yields the scale-invariant intrinsic profiles* Γ_i *and* Γ_j *(right), which have comparable ranges of values.*

in more general ways. Hence, we consider all segments to only be given by a set of segment geometries $\mathcal{S} = \{\mathbf{s}_i\}$ without connectivity or integration time information. To simplify the discussion, we assume that all curve segment $\mathbf{s}_i(s)$ are reparameterized to the uniform range $s \in [0,1]$.

Segment similarity estimation is based on comparison of profiles of individual intrinsic curve properties denoted by $\gamma(s)$ that define the extrinsic geometry up to rigid transformation. For 2D curve segments we consider signed curvatures κ_2 as their intrinsic properties γ, whereas for 3D curves we evaluate unsigned curvatures κ_3 and curve torsions $\tau(s)$. Signed curve torsion is given by

$$\tau = \frac{\det([\dot{\mathbf{c}}, \ddot{\mathbf{c}}, (\nabla\ddot{\mathbf{c}})\ddot{\mathbf{c}}])}{||\dot{\mathbf{c}} \times \ddot{\mathbf{c}}||^2} \tag{4}$$

and can be computed from vector field quantities only (cf. [WT02]). Similar to curvature, we exploit the fact that torsion scales inversely proportional to the scaling factor of a segment (see, e.g., [dC76]).

Given a pair of curve segments (not necessarily originating from the same stream line), we compute their intrinsic similarity (or distance) based on discretized profiles of intrinsic properties γ. This requires two operations: intrinsic profile discretization and profile distance computation.

2.1. Discretization of Normalized Intrinsic Property Profiles.

To enable an efficient and scale-invariant comparison of segments based on their intrinsic properties, we discretize the continuous property γ along each segment into $n > 0$ uniformly sized bins: for a segment $\mathbf{s}_i(s)$ defined over $s \in [0,1]$, the value of the k-th bin of uniform length $\frac{l_i}{n}$ corresponding to the property γ is given by the scale-normalized *average*

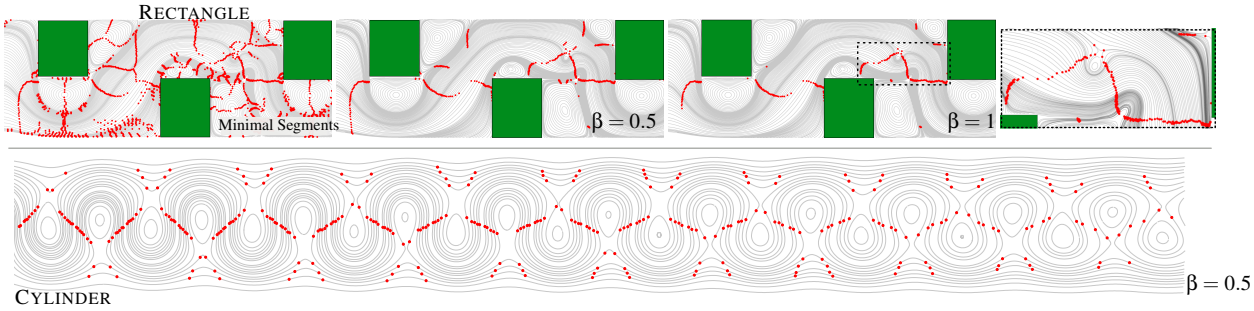

Figure 3: *Consistent Stream Line Segmentation. Starting from the minimal segments (top left) our segmentation scheme extracts segmentations for which intrinsically similar stream line segments are segmented in a compatible way. The closeup shows that segmentations form orthogonal patterns to laminar flow regions. Removal of low curvature segments is steered by the* β *parameter. The slowly varying* CYLINDER *flow on the bottom illustrates the consistency of the segmentation.*

property value

$$\gamma_k = \frac{n}{l_i^2} \int_{\frac{k}{n}}^{\frac{k+1}{n}} \gamma \, ||\dot{\mathbf{s}}_i|| \, \mathrm{d}s \, . \tag{5}$$

Note that we normalize each average property value by the segment length l_i to obtain comparable value ranges for segments of different scales. This is possible for both, curvatures and torsion, as they scale inversely proportional to the scaling factor. For the i-th segment this gives the *discretized scale-invariant intrinsic property profile* $\Gamma_i = [\gamma_0, \ldots, \gamma_{n-1}]$ of the intrinsic property γ. Scale normalization yields scale-invariant profiles that are comparable for segments of different length and scale, e.g., a scaled segment will have an *equal* profile compared to the original segment. Figure 4 exemplifies the discretization of the continuous signed curvatures of two different 2D segments. Note that even slight variations in the curve geometry result in strong intrinsic property variations that are well represented in the discretized profiles. Discretization of intrinsic properties of 3D segments along their univariate parametrization is analog. The parameter n steers the profile resolution and accuracy. In all our experiments, we observe that a value of $n = 40$ is usually sufficient to enable accurate segment comparison, e.g., for pattern retrieval. This is because our consistent stream line segmentation scheme ensures that each segment generally represents a single dominant curve feature, e.g., a single arc element, whose global intrinsic profile is similar even at lower resolutions. In addition, our segmentation guarantees that curvature profiles are bounded by absolute curvature minima, which give common reference points for the subsequent profile distance estimation.

2.2. Scale-invariant Intrinsic Profile Similarity.

Given two curve segments $\mathbf{s}_i$ and $\mathbf{s}_j$ (not necessarily originating from the same stream line), we estimate their similarity based on the distance of their normalized intrinsic profiles Γ_i and Γ_j. Both profiles have the corresponding numbers of n bins, have comparable value ranges due to scale normalization, and are approximately aligned at their bounds by minima of absolute curvature. Hence, we need to compute distances between aligned profiles with equal bin numbers but varying total sum of values.

This problem is known as histogram distance estimation, for which a number of standard alignment measures exists: Rubner et al. [RTG98] review standard bin-to-bin measures like the well known L^p distances or the χ^2 test (e.g., used in [MJL*13] for intrinsic property comparison along *whole* stream lines). They conclude that bin-to-bin measures are too sensitive to profile discretizations given by the position of bin boundaries. Instead, a cross-bin measure that is more robust w.r.t. local deformations is proposed that is based on the *Earth Movers Distance* (EMD), which is also known as transportation distance (see also [LO07]). As standard EMD is only defined for distribution-like profiles with equal total bin sums, we use the recent hEMD generalization by Pele and Werman [PW09], which is a well-defined metric also for our setting of unequal total profile sums. Intuitively, EMD measures the costs of transporting one profile to another profile if both profiles are interpreted as mass distributions. In addition, hEMD also penalizes moving mass in or out of the domain to account for unequal total masses.

Let $d(\Gamma_i, \Gamma_j)$ denote the (forward) *intrinsic scale-invariant distance* based on the profiles Γ_i and Γ_j computed by hEMD. Low values of $d(\Gamma_i, \Gamma_j)$ indicate similar intrinsic profiles. Backward distances $d_-(\Gamma_i, \Gamma_j)$ are defined by inverting the traversal order of one of the profiles. Note that hEMD is only well-defined for profiles of positive values. In case negative values are detected (e.g., for signed curvature or torsion), we shift all values of both profiles to the positive value range by adding the absolute value of the smallest negative value. This operation does not change the profile distance, which is given by their relative difference. We distinguish several cases in terms of curve dimensions and transformation invariance, which consist of invariance to rigid and similarity transformations, as well as invariance to segment orientation.

For 2D curves, we measure signed curvatures κ_2 as the intrinsic property γ. As curvatures are invariant to rigid trans-

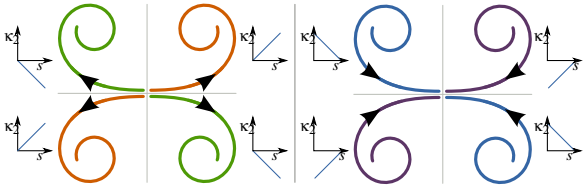

Figure 5: *Signed Curvature Correspondences. Segments are grouped by forward (left) and backward orientation (right), and for each segment the curvature profile is shown. Same-colored segments are equal w.r.t. the forward distance $d(\Gamma_i,\Gamma_j)$ (rigid transformation invariance). Axis reflexion invariance is obtained by including $d(\Gamma_i,-\Gamma_j)$. For additional orientation-invariance, $d_-(\Gamma_i,\Gamma_j)$ and $d_-(\Gamma_i,-\Gamma_j)$ need to be considered.*

formations, it is sufficient to measure the scale and *rigid transformation invariant similarity* of two segments by $d(\Gamma_i,\Gamma_j)$ using a single hEMD evaluation. Figure 5 illustrates this invariance for same-colored segments.

In addition to invariance to scale and rigid transformations, similarity transformations also allow axis reflexions. Reflexions invert the curvature profile, so we measure scale and *similarity transformation invariant similarity* by

$$\min(d(\Gamma_i,\Gamma_j), d(-\Gamma_i,\Gamma_j)) \,, \tag{6}$$

i.e., by the smallest distance that also permits reflexions, which requires two hEMD evaluations. Here, we denote an inverted intrinsic profile by $-\Gamma_i = [-\gamma_0,\ldots,-\gamma_{n-1}]$. Note that profile distance is symmetric w.r.t. profile inversion, i.e., $d(-\Gamma_i,\Gamma_j) = d(\Gamma_i,-\Gamma_j)$. Both groups of segments in Figure 5 correspond to segments that are invariant under similarity transformations w.r.t. this measure.

Finally, segment curvature still depends on the flow aligned orientation of the original stream lines. If orientation should not be respected by the similarity estimation, we use the scale, *orientation, and similarity transformation invariant similarity*

$$\min(d(\Gamma_i,\Gamma_j), d(-\Gamma_i,\Gamma_j), d_-(\Gamma_i,\Gamma_j), d_-(-\Gamma_i,\Gamma_j)) \tag{7}$$

by considering two additional backward differences, which invert profile orientation. W.r.t. this most general similarity all segments in Figure 5 are equal. It requires four hEMD evaluations. This measure is also useful if no initial curve orientation is defined, e.g., for curves integrated in orientation-free direction fields [CZCE08, JDL09].

For 3D segment similarity estimation, we combine differences in unsigned curvature κ_3 and torsion τ: 3D similarity is given by the weighted sum $d_{\kappa_3}(\Gamma_i,\Gamma_j) + w_\tau d_\tau(\Gamma_i,\Gamma_j)$ of individual profile distances in curvature $d_{\kappa_3}(\Gamma_i,\Gamma_j)$ and torsion $d_\tau(\Gamma_i,\Gamma_j)$. Our experiments indicate that torsion generally has a much stronger value variation than curvature as it is a third order segment quantity. Hence, usually a weight parameter $w_\tau < 1$ is chosen to reduce the influence of torsion to the final similarity estimation. To compute $d_\tau(\Gamma_i,\Gamma_j)$ for a given transformation invariance type, note that torsion is a signed quantity similar to 2D curvature κ_2. Therefore, the same rules for rigid, similarity, and orientation-invariance apply and require a single, two, or four hEMD evaluations. In addition, as the unsigned curvature is invariant to profile inversion, orientation invariant similarity estimation requires *only* two hEMD evaluations. Hence, for this most general similarity estimation, six hEMD evaluations are required for 3D segments.

References

[CZCE08] CHEN W., ZHANG S., CORREIA S., EBERT D. S.: Abstractive representation and exploration of hierarchically clustered diffusion tensor fiber tracts. *Comput. Graph. Forum (Proc. EuroVis) 27*, 3 (2008), 1071–1078. (Cited on page 5)

[dC76] DO CARMO M. P.: *Differential Geometry of Curves and Surfaces*. Prentice–Hall, 1976. (Cited on pages 1, 2, and 3)

[JDL09] JIANU R., DEMIRALP C., LAIDLAW D. H.: Exploring 3d dti fiber tracts with linked 2d representations. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1449–1456. (Cited on page 5)

[LO07] LING H., OKADA K.: An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence 29*, 5 (2007), 840–853. (Cited on page 4)

[MJL*13] MCLOUGHLIN T., JONES M. W., LARAMEE R. S., MALKI R., MASTERS I., HANSEN C. D.: Similarity measures for enhancing interactive streamline seeding. *IEEE Transactions on Visualization and Computer Graphics 19*, 8 (2013), 1342–1353. (Cited on pages 2 and 4)

[PW09] PELE O., WERMAN M.: Fast and robust earth mover's distances. In *Proc. ICCV* (2009), IEEE. (Cited on page 4)

[RTG98] RUBNER Y., TOMASI C., GUIBAS L.: A metric for distributions with applications to image databases. In *Proc. ICCV* (1998), pp. 59–66. (Cited on page 4)

[WT02] WEINKAUF T., THEISEL H.: Curvature measures of 3D vector fields and their applications. *Journal of WSCG 10*, 2 (2002), 507–514. (Cited on pages 1 and 3)