PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Topology-based methods for quantitative comparisons of vector fields

Rajesh Batra, Kerstin Kling, Lambertus Hesselink

Rajesh Batra, Kerstin Kling, Lambertus Hesselink, "Topology-based methods for quantitative comparisons of vector fields," Proc. SPIE 3960, Visual Data Exploration and Analysis VII, (28 February 2000); doi: 10.1117/12.378904

SPIE.

Event: Electronic Imaging, 2000, San Jose, CA, United States

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019 Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

Topology Based Methods for Quantitative Comparisons of Vector Fields

Rajesh Batra[a], Kerstin Kling[b] and Lambertus Hesselink[c]

[a]Dept of Aeronautics/Astronautics, Stanford University, Stanford, CA 94305
[b]Institute of Process Engineering, University of Hannover, Hannover, Germany D-30167
[c]Dept of Electrical Engineering, Stanford University, Stanford, CA 94305

ABSTRACT

Computers have enabled the generation and acquisition of large vector and tensor data-sets. Enormous progress has been made in modeling complex systems such as the flow over airplanes with realistic Reynolds numbers, and in data acquisition producing large data-sets such as weather and ocean current databases. However, techniques for automated searching and comparison of large vector and tensor data-sets have been lacking. Many of the techniques are based on visual comparisons which do not have the quantitative rigor required for automated comparisons. Other methods which use data based comparisons require a common domain, grid alignment, and interpolation often making data-sets acquired from different sources difficult to compare.

This paper provides comparison methods that are based on the topology of the vector field. These methods can be used for both 2-D and 3-D fields, and depending upon the complexity of the field, different levels of comparisons can be utilized to reduce search times. Topology based methods have the added benefits of describing a field that is independent of the grid, and that is based solely on topological components which have a significant compression over the original field.

Keywords: Vector Fields, Comparison, Graph Matching, Earth Mover's Distance

1. INTRODUCTION

There has been a steady increase in the need for field comparisons. Predominantly, techniques have been developed for scalar field comparisons in such applications as image retrieval systems found in many popular web search engines,[1] and in security systems utilizing face recognition.[2] Increasing the order of the field to multivariate systems increases the complexity of the comparison problem and therefore few techniques have been developed to address this issue. In the physical sciences, vector fields are a predominate quantity used in the modeling and analysis of dynamical systems, electromagnetism, and fluid mechanics. Accurately comparing vector fields aids in the analysis and understanding of the physical phenomena. For example, it is often necessary to validate simulated flow fields with experimentally acquired fields.

A variety of comparison techniques exist for vector fields and are categorized as image, data, or feature based comparisons. Image based techniques convert numerical data into computer generated images and rely typically on side-by-side comparisons.[3] This category of techniques often lack autonomous means for comparisons and are more qualitative in nature. Data level comparison techniques operate directly on the raw data. An accurate comparison requires proper grid alignment which can involve problematic interpolation between two fields.[4] Feature based comparisons extract problem specific phenomena often represented through geometry such as a shock surface in a 3-D flow field. Since features are problem specific, typically expertise within the area is necessary for extracting and understanding the comparison. The features may have a semantic representation which allows for autonomous comparison; however, a survey of the predominant methods indicate the comparison is typically qualitative in nature. To summarize, the majority of techniques available for vector field comparisons lack quantitative rigor, and/or suffer from alignment issues which prevent general comparisons of fields acquired through different techniques, and/or lacks autonomy which prevents the creation of vector field databases such as a database of weather patterns.

Further author information: (Send correspondence to R.Batra)
R.Batra: E-mail: raj@kaos.stanford.edu
K.Kling: E-mail: Kerstin.Kling@stud.uni-hannover.de
L.Hesselink: E-mail: bert@kaos.stanford.edu

In *Visual Data Exploration and Analysis VII*, Robert F. Erbacher, Philip C. Chen, Jonathan C. Roberts, Craig M. Wittenbrink, Editors, Proceedings of SPIE Vol. 3960 (2000) • 0277-786X/00/$15.00

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

This paper introduces a new method for comparing flow fields by utilizing the topological structure of the field. Briefly, vector field * topology is conceptualized as a schematic depiction of the behavior of a large collection of streamlines. The basic constituents are critical points and the set of connecting streamlines.[5] The topological structure forms a skeleton that schematically describes the flow field globally. Rather than compare the field on a vector by vector basis, the topological constituents are compared. This avoids alignment of the field and is grid independent. Furthermore, by representing the flow field as a topological skeleton, a sizable compression of the field is realized. This not only allows for large databases to exist, but enables high speed, automated searches. Obviously, in order to execute an autonomous search a quantitative metric needs to be established to compare topological fields. This metric first introduced by the authors Lavin, Batra, and Hesselink in[6] describes the critical points in a metric space. It is further extended in this paper by considering the connections between critical points which represents the flow field as a graph thereby improving the description of the vector field. Two error correcting graph matching algorithms are introduced. The first being a conventional $A\star$ algorithm, and the second though not optimal is highly efficient since it utilizes earth mover's distance, a simplex based algorithm. The paper is concluded by proposing a method to classify higher order critical points within the flow field.

2. REPRESENTATION OF CRITICAL POINTS IN A METRIC SPACE

A 2-D vector field which is assumed to be continuous and have continuous partial derivatives can be described as a system of two simultaneous differential equations. A parametric solution to this system can be plotted as a phase portrait and is discussed in many differential equation texts.[7] The topology of the phase portrait is composed of critical points, which are points in the flow where the magnitude of the vector field vanishes and where the slope of the streamlines is locally indeterminate, and the set of connecting streamlines which originate and terminate at critical points.

A critical point is termed isolated or simple if an open neighborhood exists such that no other critical point lies within this neighborhood, and the critical point can be described using first order approximations. In section 5, higher order critical points will be considered. The behavior of the flow about a critical point is analyzed by investigating the trajectories in the neighborhood of the critical point. Equation 1 describes the vector field that is a distance dx, dy away from the critical point. The distance is assumed sufficiently small such that only first order terms are retained.

$$v_x(dx,dy) \approx \frac{\partial v_x}{\partial x}dx + \frac{\partial v_x}{\partial y}dy \qquad v_y(dx,dy) \approx \frac{\partial v_y}{\partial x}dx + \frac{\partial v_y}{\partial y}dy \tag{1}$$

The flow pattern is completely determined by the Jacobian, $J_{ij} = \frac{\partial v_i}{\partial v_j}$ $(i,j = 1,2)$ evaluated at the critical point. The visualization and categorization of the flow patterns have traditionally been accomplished by analyzing the eigenvalues of the characteristic equation:

$$\lambda^2 + P\lambda + Q = 0 \tag{2}$$

where $P = -trace(J)$ and $Q = det(J)$ which forms the well known $P - Q$ stability diagram.[8] This diagram is depicted in Figure 1(a). Note that the various flow patterns are determined by the sign of P, and the discriminant of the characteristic equation $D = P^2 - 4Q$. A benefit of this space is that a simple computation readily maps to a flow pattern solution. However, given two critical points (P_0, Q_0), and (P_1, Q_1) there is an inability to compare how close these flow patterns are to one another. The assumption is that like patterns are considered closer to one another. The parabola readily portrays the non-linearities in this space.

A new space, (α', β'), as explained in[6] is defined such that $\alpha = P$ and $\beta = sign(P^2 - 4Q)\sqrt{|P^2 - 4Q|}$ and are normalized as follows:

$$\alpha' = \frac{\alpha}{\sqrt{\alpha^2 + \beta^2}} \qquad \beta' = \frac{\beta}{\sqrt{\alpha^2 + \beta^2}} \tag{3}$$

The mapping is shown in Figure 1(b). Note that the flow pattern is completely described by the ratio of β' to α', i.e. an angle. The origin $(\alpha' = 0, \beta' = 0)$ corresponds to uniform flow. The categorization of the flow field using a single value, namely the angle, has many immediate benefits:

*The definition of vector fields is restricted to continuous fields or flows.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

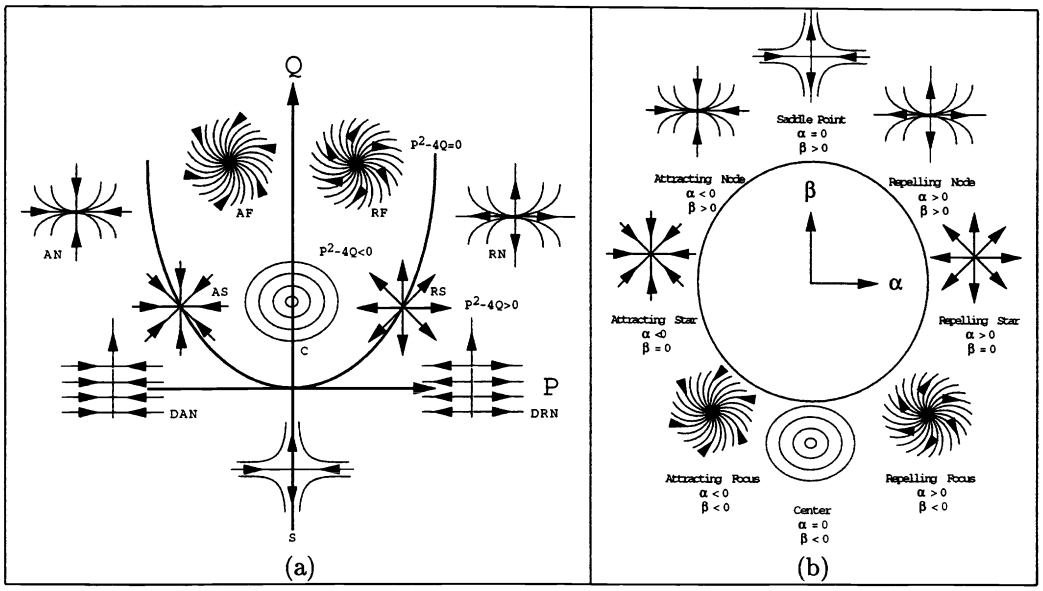

Figure 1. (a) Simple critical points mapped in $P-Q$ space. DRN: Degenerate Repelling Node, RS: Repelling Star, RN: Repelling Node, RF: Repelling Focus, C: Center, AF: Attracting Focus, AS: Attracting Star, AN: Attracting Node, DAN: Degenerate Attracting Node, S: Saddle. (b) Simple critical points mapped into $\alpha' - \beta'$ space.

1. The change in pattern is seen as a continuous transformation in the α', β' space parameterized by the angle. For example, bifurcation is geometrically represented as the phase transition through $\theta = 45^o$ or $\theta = 135^o$ (from a node to a saddle). The structural instability of non-hyperbolic critical points is apparent in this space as they are represented as single points (one angle) as opposed to hyperbolic fixed points which span a range of θ. Slight perturbations due to neighboring critical points, or higher order terms can transform these points. For example, a center with higher order terms can transition to either an attracting or repelling focus.

2. Certain physical phenomena can easily be identified. For example, the separation points are immediately identified as the transition in color in the flow over a circular cylinder in Figure 2(a). Since the flow is incompressible $\alpha = 0$, only saddles and centers can exist about the critical points ($\beta = \pm 1$). Close to the body there are regions of high circulation due to viscosity and is indicated by the darker zones ($\beta = -1$), and at the separation points, the color changes (lighter) as they are typically described as half-saddles.[9]

3. In this space, critical points obey all the rules defined for a regular 2-D Euclidean space and the distance between any two critical points is a metric. The next section uses this property for comparing vector fields.

3. CRITICAL POINT MATCHING USING EARTH MOVER'S DISTANCE

Earth mover's distance (EMD), introduced by Rubner, Guibas and Tomasi, is a metric between two distributions that was used for fast content-based image retrieval for large database applications.[11] Images were described by amounts and distribution of color and texture. EMD computed the minimal cost to transform one distribution into another under a set of constraints. In a similar vein, a feature distribution can be defined for a vector field. Details of the distribution definition can be found in reference.[6] An overview of the method is briefly discussed followed by a benchmark study to investigate the method's utility for larger problems (i.e. more critical points) as in the case for turbulent flows.

The EMD algorithm requires a feature distribution and a cost function definition. For a vector field, the feature distribution is the set of (α, β) values that represent the critical points. The cost function is the Euclidean distance between the (α, β) coordinate points, i.e. the cost to convert critical point i to critical point j is

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

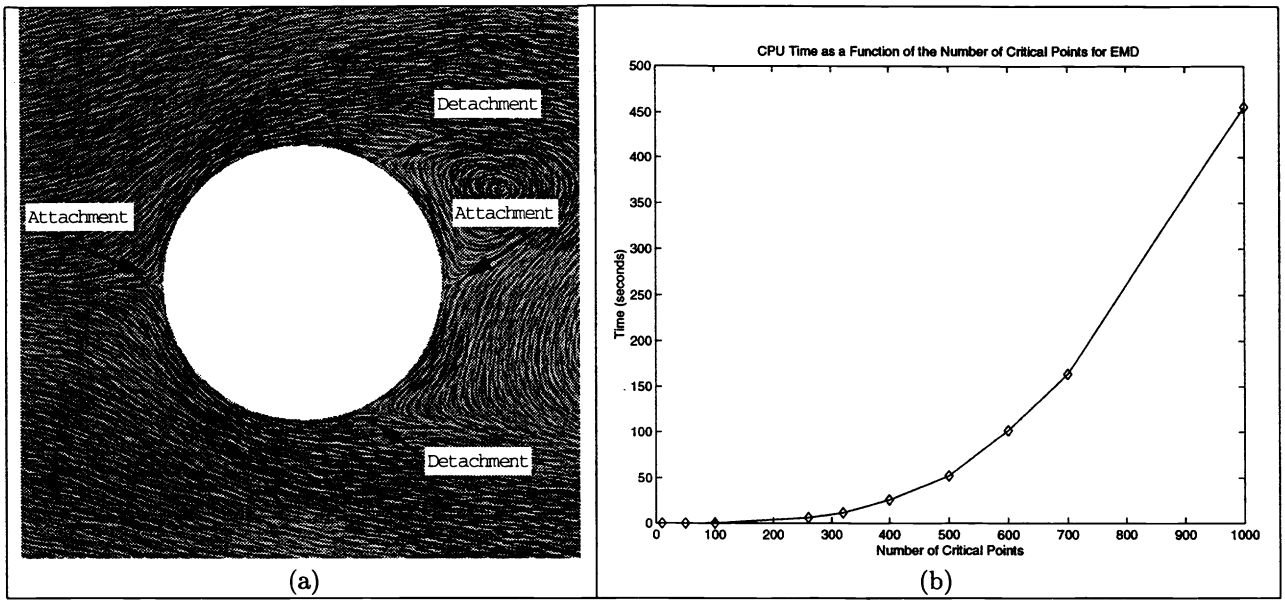

Figure 2. (a)Separation points can quickly be identified by mapping the scalar $\theta = \tan^{-1}\left(\frac{\beta'}{\alpha'}\right)$. Flow over a cylinder, Re=200 viscous, incompressible.[10] (b) Time required to compare fields with equal number of critical points using EMD. 5 cases were run, and averaged to produce plot.

$d_{ij} = \sqrt{(\alpha_i - \alpha_j)^2 + (\beta_i - \beta_j)^2}$. Therefore, the cost required to transform a vector field with distribution

$$\{(\alpha_1, \beta_1), (\alpha_2, \beta_2), \ldots, (\alpha_n, \beta_n)\}$$

to a vector field with distribution

$$\{(\alpha'_1, \beta'_1), (\alpha'_2, \beta'_2), \ldots, (\alpha'_n, \beta'_n)\}.$$

is $Cost = \sum_{i=1}^{n} \sqrt{((\alpha_i - \alpha'_i)^2 + (\beta_i - \beta'_i)^2)}$. EMD minimizes the total $Cost$ by computing a one-to-one mapping of the (α, β) values in field 1 to the (α, β) values in field 2. The connections between the points are ignored, only the distribution of critical points is considered. Past results have shown that the measurement above is effective in finding regularity in the flow fields and as a diagnostic tool for validating simulations .[12] Previous field comparisons have focused on laminar flows with a small number of critical points (typically under 10). More complex flows have more critical points, therefore an important area to analyze EMD's effectiveness. The EMD algorithm is based on the transportation problem in linear programming and uses the simplex method to solve the system.[13] The simplex method is known to run in exponential time, however in practice it typically runs more efficiently.[14] Figure 2(b) measures the performance of the EMD algorithm applied to vector field comparisons. The tests were performed on a sgi 2 processor Onyx 2 running at 195 Mhz on MIPS R10000 processors. The time required to extract the critical points was not included, as this was considered a preprocessing step. The vector field database contains only topological information. Critical points varying from 10 to 1000 were compared. Only equal number of critical points were compared to one another. The fields were generated randomly, and the tests were computed 5 times and then averaged. Even up to a 100 critical points, EMD required under 0.25 seconds to solve the system. The performance begins to noticeable degrade at about 300. Nonetheless, the time scale approaches minutes and not hours as is the case for graph methods to be discussed in section 4. Additionally, the number of EMD comparisons can greatly be reduced in a large database search by first computing the lower bound of the EMD and culling those fields that do not meet the minimum distance measure. Details are described in.[15]

A second test was performed to see if complicated flow patterns could be distinguished. In the past, flow fields with no more than 10 critical points were used for the comparison technique. The viscous, incompressible flow past a long, square cylinder with an angle of attack of 0^o at Re=22,000 is highly turbulent with complex physical interactions of

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

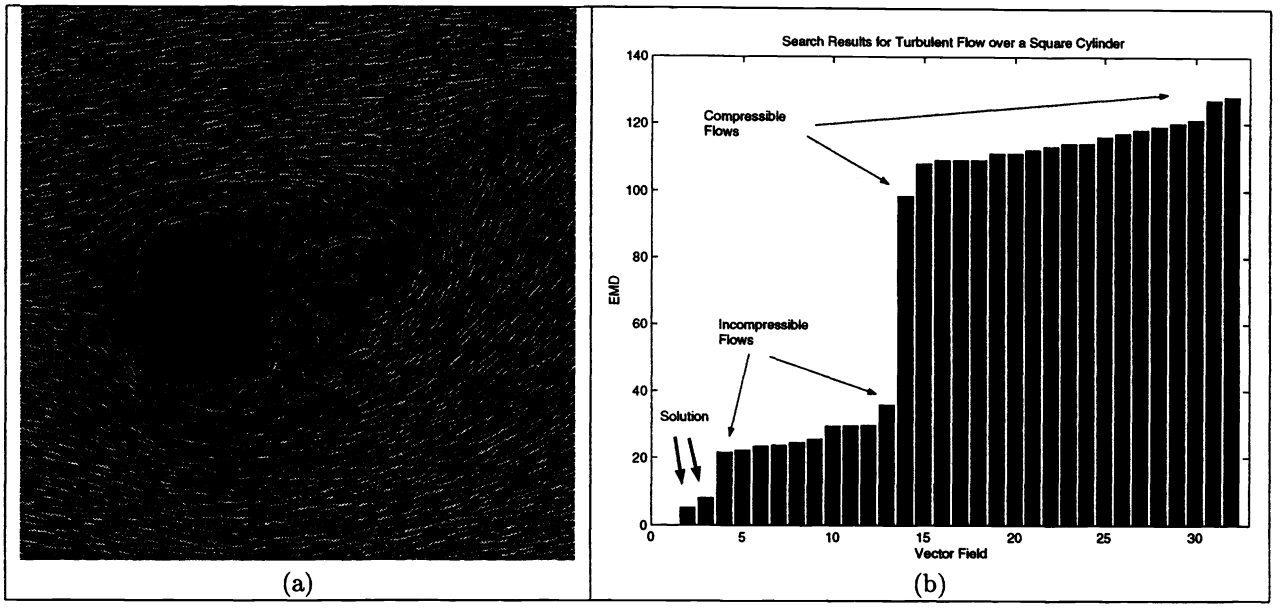

Figure 3. (a)Flow over a square cylinder, Re=22,000 viscous, incompressible,[16] searched for in a database. (b) Database search results sorted by EMD. Bars marked solution with lowest energy (closest match) are in fact the fields computed in the same simulation as (a) but a few times steps later.

swirling flows combined with physical phenomena such as transition, separation, attachment/detachment, and vortex shedding. Figure 3(a) depicts the cross section of the flow about the block. The data was generated by R.W.C.P. Verstappen and A.E.P. Veldman of the university of Groningen; details of the simulation can be found in .[16] The flow contains on the order of 260 critical points which outwardly appear to be randomly distributed by type. To test whether this is the case, 29 additionally fields were randomly created which contain the same order of critical points (between 255-270) as a representative vector database. Since the simulation in[16] was for incompressible flow, several of the 29 randomly generated fields mimicked the flow patterns found in incompressible flow to further challenge the comparison. Finally, two additional fields were added to the database that were computed a few time steps away from Figure 3(a). The expectation is that if the flow in Figure 3(a) is searched for in the database the closest match would be the two fields that were computed a few time steps away.

Figure 3(b) depicts the EMD sorted results of searching for the field in Figure 3(a). The two fields that simulated the flow over the square cylinder did return as the best match and are marked as *solution* in the figure. After that, the fields with topological structures that represent incompressible flows were found. There is a marked increase between the incompressible flow and the matched values indicating a good discriminant. Finally, an even more significant jump in EMD represents the furthest match which is that of the compressible flows. The measure can be further improved by considering the connections between critical points. This is the topic of the next section.

4. GRAPH METHODS

In the previous section, comparisons were accomplished using only the distribution of critical points in a flow field. Due to the inherent restraints in critical point interactions, it is possible for a velocity field to have a unique signature described only by its critical points.[8] However, this is not always the case. A vector field's representation can be improved if the streamlines that interconnect between critical points is considered. The combination of critical points and connecting streamlines forms an attributed, relational graph (ARG). This type of graph is composed of a set of property based nodes which are the critical points described by the coordinate pair (α, β), and relations between the nodes represented by edges which are the connecting streamlines. For two graphs to be isomorphic, there must be a one-to-one correspondence between their node sets which preserves adjacency. To compare graphs error correction occurs such that graph G_i becomes isomorphic to graph G_j. The cost to transform G_i to G_j is defined through a similarity or distance measure $D(G_i, G_j)$. This measure can be thought of as the composition of two components:

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

(1) The cost of transforming the properties of the nodes and (2) the cost to match the relations (adjacencies) of the graph. Both operations are discussed in the following paragraphs.

1. The process of transforming the properties of a node (node-matching) can be decomposed into three edit operations: (a) insert a node: x_{ni} (b) delete a node: x_{nd} (c) convert a node p to a node q: $x_{nt}(p,q)$
2. The process of matching edge relations for two graphs can be decomposed into a similar set of edit operations. (a) insert an edge: x_{ei} (b) delete an edge: x_{ed} (c) convert edge e_p to edge e_q: $x_{et}(e_p,e_q)$

A graph transformation can be represented by a sequence of edit operations. As long as the graph to be transformed is not affected by an edit operation more than once, the sequence of edit operations can be divided into the following sequences[17]: Let $X_{ni} = \{x_{ni}^{(1)}, x_{ni}^{(2)}, \ldots, x_{ni}^{(n_1)}\}$ represent the sequence of n_1 node insertion operations. Similarly, $X_{nd} = \{x_{nd}^{(1)}, x_{nd}^{(2)}, \ldots, x_{nd}^{(n_2)}\}$, $X_{nt} = \{x_{nt}^{(1)}(p_1,q_1), x_{nt}^{(2)}(p_2,q_2), \ldots, x_{nt}^{(n_3)}(p_{n_3},q_{n_3})\}$, $X_{ei} = \{x_{ei}^{(1)}, x_{ei}^{(2)}, \ldots, x_{ei}^{(n_4)}\}$, $X_{ed} = \{x_{ed}^{(1)}, x_{ed}^{(2)}, \ldots, x_{ed}^{(n_5)}\}$, and $X_{et} = \{x_{et}^{(1)}(e_{p_1},e_{q_1}), x_{et}^{(2)}(e_{p_2},e_{q_2}), \ldots, x_{et}^{(n_6)}(e_{p_{n_6}},e_{q_{n_6}})\}$ represent the sequence of node deletion, node conversion, edge insertion, edge deletion and edge conversion operations, respectively. With each edit operation, there is an associated cost. The total cost of all edit operations,$X = X_{ni} + X_{nd} + X_{nt} + X_{ei} + X_{ed} + X_{et}$, can be defined as follows:

$$C(X) = w_{ni}\sum_{i=1}^{n1} c(x_{ni}^{(i)}) + w_{nd}\sum_{i=1}^{n2} c(x_{nd}^{(i)}) + w_{nt}\sum_{i=1}^{n3} c(x_{nt}^{(i)}(p_i,q_i)) \quad + \tag{4}$$
$$w_{ei}\sum_{i=1}^{n4} c(x_{ei}^{(i)}) + w_{ed}\sum_{i=1}^{n5} c(x_{ed}^{(i)}) + w_{et}\sum_{i=1}^{n6} c(x_{et}^{(i)}(e_{p_i},e_{q_i}))$$

where w is an associated weight factor. The sum of all weights must equal 1. The distance measure for graph isomorphism is the minimum cost over all edit operations:

$$D(G_i,G_j) = min_X C(X) \tag{5}$$

For vector field comparisons, the cost to transform node p_i to node q_i is the Euclidean distance between the respective (α,β) coordinates. Rather than include node insertion and deletion operations in the distance measure, normal points $(\alpha,\beta) = (0,0)$ are added in the preprocess phase such that the number of critical points in the two fields to be compared are the same. The node transform operator can convert a normal point to any other critical point. By removing node insertion and deletion operations, the performance is increased as there are two less operations to minimize over. All edges (streamlines) are assumed equal, therefore edge transformations are not required, only edge insertion and deletion. The cost associated with edge insertion/deletion is unity, and was selected as such to be consistent with the fact that the cost to convert a normal point to any other critical point is unity.

4.1. Optimal Error Correcting Graph Isomorphism Algorithms Applied to Vector Fields

To compute the distance measure stated in equation 5, an optimal algorithm is required such that the minimum cost in terms of edit operations is guaranteed to be found. Unfortunately, error correcting (sub)graph isomorphism is an NP complete problem, i.e. no algorithm can be constructed that guarantees to find an optimal isomorphism in polynomial time.[18] The simplest algorithm for optimal graph matching is to represent a graph using an adjacency matrix. This nxn matrix where n is the number of nodes (critical points) is assigned a value of 1 if critical point i is connected to critical point j, otherwise a 0 is assigned. To compare graph G_i to G_j, the corresponding adjacency matrix is compared component wise. The minimum is found by computing the difference between all possible permutations of the adjacency matrix and selecting the minimum. The algorithm as applied to vector fields is discussed in.[19,20] This algorithm works in a reasonable amount of time for simple flows up to 10 critical points as the complexity of the algorithm is $O(n!n^2)$. A typical time on the Onyx 2 used above for 7 critical points is 60 ms, for 9 critical points about 1 minute, and for 10 critical points (3.6 million permutations) about 15 minutes.

For vector fields with greater than 10 critical points, a different approach is required. One of the most popular family of algorithms for optimal, error correcting graph isomorphisms is based on the $A\star$ algorithm.[21] This branch and bound problem can be described as follows. An input graph of size m is to be compared to a model graph of size n, where $m \leq n$. For vector field comparisons, $m = n$, but if partial matches (subgraph isomorphism) are allowed

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

```
que->push(ROOTNODE)
do                                  /* que is a priority que which places
    if que->empty() return(FAIL)     * elements (via push()) with least cost
    node = que->pop()                * at the front of the que, internally
    if (node->goal()) return(node)   * calling node->cost() for each child.
    que->push(node->children())      */
end
```

Table 1. Pseudo code for general $A\star$ algorithm.

$m < n$ is acceptable. A decision tree can be formulated by sequentially comparing the vertices in the input graph to the vertices in the model graph. The tree has height m, and $n - p$ children at level $p = 0, 1 \ldots m - 1$. The levels of $p > 0$ correspond to the sequential traversal of the input graph. At a given level i,$i = 1 \ldots p$, let N denote a unique node-match path from the root to i, i.e. $N = \{(1, q_1), (2, q_2) \ldots (i, q_i)\}$. At each node, N, there is an associated cost which is the determining factor as to whether the node is further expanded. For the $A\star$ algorithm, the cost is the sum of the current path cost $g^*(N)$ and the cost to the goal $h^*(N)$ which is approximated by an admissible heuristic (cost less than $h^*(N)$), $h(N)$. $g^*(N) = \sum_{i=1}^{p} k(i, q_i)$ where k is the cost to convert node i to node q_i and the cost to convert all edges connected to i from vertices already enumerated (values $< i$) in the graph to corresponding edges in the model connecting to q_i. Mathematically,

$$k(p, q_p) = w_{nt} c(p, q_p) + w_{et} \sum_{j=1}^{p-1} c((j, p), (q_j, q_p)) \tag{6}$$

where $c((j,p),(q_j,q_p))$ is the cost to convert edge (j,p) to edge (q_j,q_p), w_{nt} and w_{et} are optional weighting factors. See[22,21] for further details. The computation of $h(N)$, the heuristic function to estimate the cost remaining, has no one solution. Depending on how close $h(N)$ is to $h^*(N)$ (the exact future cost), the number of nodes to expand can greatly be reduced. $h(N) = 0$ corresponds to uniform cost search[23] which expands the greatest number of nodes for an optimal and complete algorithm. Wong et. al. proposed a straight forward heuristic to significantly reduce the number of nodes to expand. The future cost to the goal is estimated by aggregating the minimum costs of transforming the remaining vertices of the input graph to the model. Since this is only an estimate, a many to one mapping is possible; however, this guarantees that the estimated cost is below the actual future cost. Formally, let M_1 be the remaining unmatched vertices in the input and M_2 be the unmatched vertices in the model. Therefore,

$$h(N) = w_{nt} \sum_{i=p+1}^{m} \min_{q \in M_2} (k'(i, q)) + w_{et} \sum_{\substack{ij \in M_1 \\ i<j}} \min_{\substack{q \neq r \\ q,r \in M_2}} (c((i, j), (q, r))) \tag{7}$$

where $k'(i,q) = c(i,q) + \sum_{j=1}^{p} c((i,j),(q,q_j))$. For vector fields, since edges have no attributes, the edge cost is 0 if M_1 and M_2 both do or do not contain edges, else cost is 1. The search through this state space can succinctly be described by Table 1.

A simple example will illustrate the algorithm. Figures 4(a)(c) depict two similar vector fields each containing 1 saddle, 2 attracting foci, and 2 repelling foci points. Figures 4(b)(c) depict the corresponding graphs with the associated (α, β) values that describe the critical points. The decision tree shown in Figure 5(a) starts by expanding the root node into 5 nodes where each node in the model is matched to the first vertex of the input. For example, RF^0 in the input matches RF^0 of the model. The cost, $f(N) = g^*(N) + h(N)$ is computed as follows: $g(N) = k(RF^0, RF^0) = 0.5 * (\sqrt{(.477 - .782)^2 + (-.879 + .623)^2} + 0) = 0.199$, assuming 0.5 is the weight. The future cost is computed by matching the remaining nodes in the input graph $\{AF^1, S^2, AF^3, RF^4\}$ to the model graph,$\{AF^1, S^2, AF^3, RF^4\}$ using the lowest cost. This operation has quadratic complexity, in this case the match can be seen by inspection: $h(N) = 0.5[.143(AF^1 \rightarrow AF^1) + .238(S^2 \rightarrow S^2) + 0(AF^3 \rightarrow AF^1) + .412(RF^4 \rightarrow RF^4)] = .396$. Note that AF^1 and AF^3 map to the same model critical point. The entire comparison requires 7 nodes to be expanded. A uniform search would have expanded 18 nodes. $A\star$'s performance is highly dependent on the size and

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

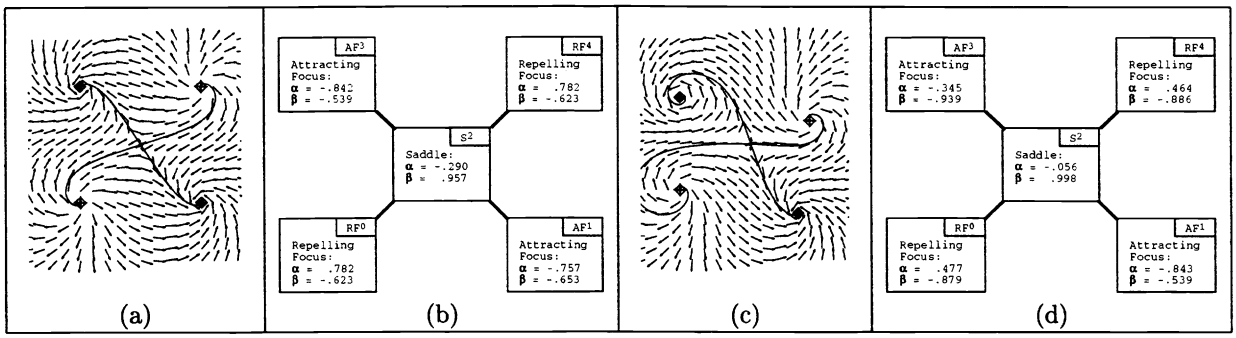

Figure 4. (a)(c) Vector field containing 5 critical points and the corresponding separatrices integrated out from the critical points. (b)(d) Attributed relational graph representation of vector fields (a)(c).

number of errors in the graphs. Both the time and space requirements for $A\star$ is exponential. Improvements have been made on the space requirements for $A\star$ such as $SMA\star$ (simplified memory-bounded) and $IDA\star$ (iterative deepening) which require space that is either customizable or proportional to the number of critical points and the depth to the solution.[23] In the best case, the time complexity of $A\star$ with future cost heuristic by Wong et al. is $O(n^2m^2)$ and in the worst case $O(n^2m^{n+1})$[21] where m and n are the number of critical points in the input and model graph. Depending on the number of errors between two graphs, a problem with 50 critical points can be solved in less than 25 seconds to several days (on the Onyx discussed above). EMD can perhaps be used in selecting graphs that are highly similar prior to running an optimal algorithm. If errors can be tolerated, non-optimal methods have much better performance and is the topic of the next section.

4.2. Non-optimal Error Correcting Graph Isomorphism Algorithms Applied to Vector Fields

Several algorithms which achieve polynomial time performance for error correcting graph matching are available at the expense of not locating the optimal solution. $D(G_i, G_j)$ is a metric only if the minimum solution is found; however, depending on the application this may not be necessary. Kittler et al. compute the maximum probability for graph matching.[24] Neural network approaches have successfully been applied.[25,26] Other methods have used eigenvalue decomposition[27] and linear programming methods.[28]

A readily extensible approach based on EMD is available with similar performance that considers the connections between critical points. A vector field with n critical points can be decomposed into n tree structures of depth 2. The tree consists of a critical point and the immediate set of connections. The comparison occurs at two levels. The first is at a global level where EMD minimizes the matching of all critical points which are now represented as tree structures. When a critical point is compared to another critical point, the total cost is the cost required to convert the head of the of the critical point and the minimum cost required to connect the children. The cost of the children is computed at the second level by again running EMD. For example, the graph in Figure 4(b) can be represented by 5 tree structures as shown in Figure 5(b), similarly for Figure 4(b). Earth Mover's Distance is used to minimize the cost between tree structures that represent the two fields. Upon computing the cost to convert S^2 in field one to field 2, EMD is again employed as shown in Figure 5(c). The total cost to convert the S^2 tree in field 1 to S^2 in field 2 is

$$cost(\underbrace{S^2_{tree}}_{field\ 1} \rightarrow \underbrace{S^2_{tree}}_{field\ 2}) = \underbrace{S^2}_{field\ 1} \rightarrow \underbrace{S^2}_{field\ 2} + EMD(\underbrace{\{RF^0, AF^1, AF^3, RF^4\}}_{field\ 1} \rightarrow \underbrace{\{RF^0, AF^1, AF^3, RF^4\}}_{field\ 2}) \quad (8)$$

where $S^2 \rightarrow S^2$ is the Euclidean distance between (α, β) values. Since the EMD and Euclidean distances are both metrics, it can be shown that this advanced EMD (AEMD) is also a metric, since it is the sum of two metrics. Detailed definitions of the AEMD signature can be found in reference.[19] Since AEMD uses only the immediate connections to estimate a graph, it is possible to produce false positives.[20] However results for vector fields have shown in practice to be comparable to optimal graph methods.[19] AEMD's lower bound will perform as well as EMD when no connections are present. The performance is highly dependent on the total number of critical points and the

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

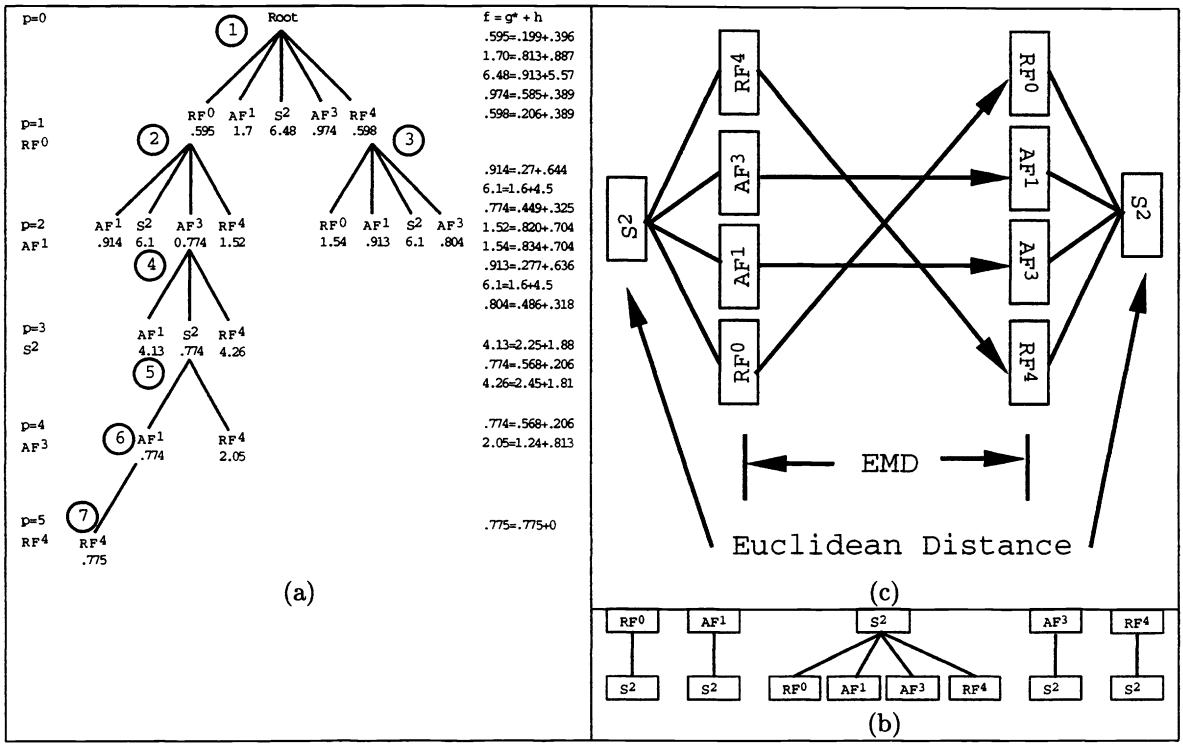

Figure 5. (a) $A\star$ algorithm with future cost applied to comparing vector fields in Figure 4(a)(c). (b) AEMD tree representation of fields in Figure 4(a)(c) (c) Schematic representation of AEMD comparison.

number of connections between critical points. For vector field topology, using simple critical points, a saddle point can have up to a maximum of 4 connections. Hence, as shown in Figure 2(b), the time required to compute the EMD of 4 critical points is under 10 ms. Figure 6(a) depicts a plot similar to Figure 3(b). In this case, the AEMD value is computed to locate the turbulent flow field shown in Figure 3(a). Since connection information is now incorporated, the solutions have much lower values than the other vector fields hence being an improved discriminant than EMD alone. This paper has focused on isolated critical points in 2-D. The methods described can be extended to 3-D and are discussed in reference.[29] The next section proposes a method to extend these techniques to higher order critical points.

5. COMPARISON OF HIGHER ORDER CRITICAL POINTS

A higher order critical point is a nonlinear, isolated zero point with multiplicity greater than 1. Typically the index or winding number (which is the number of counterclockwise or clockwise revolutions on a closed path around the critical point) is less or greater than 1. There are higher order critical points that have indices of 0,1,-1 but are not currently considered. Figure 6(b) depicts a higher order critical point classified as a monkey saddle which is composed of two super positioned saddle points. Since higher order critical points can have arbitrarily large multiplicities, it is possible to have arbitrarily large number of classifications. Furthermore, it is not apparent how these fields would fit into the (α, β) space and remain a metric.

Instead of attempting to classify higher order critical points with higher order Taylor series expansions, the approach will be to decompose a higher order critical point into a set of irreducible first order critical points. A vector field in the region of a higher order critical point will be modeled as a polynomial of order n which can be factored into n components. These components can be classified as simple critical points in (α, β) space. Scheuermann et al. simplified the representation and composition of vector fields using a four-dimensional Clifford algebra .[30,31] Succinctly, Let $E : \mathbf{C}^2 \rightarrow \mathbf{C}$ be a complex polynomial so that $\vec{v}(\vec{r}) = E(z, \bar{z})e_1$. Let $F_k : \mathbf{C}^2 \rightarrow \mathbf{C}$, $k = 1, \ldots, n$ be the irreducible, components of E so that $E(z, \bar{z}) = \prod_{k=1}^{n} F_k$. Then the vector fields $\vec{w_k}(\vec{r}) = F_k e_1$ have isolated zeros

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

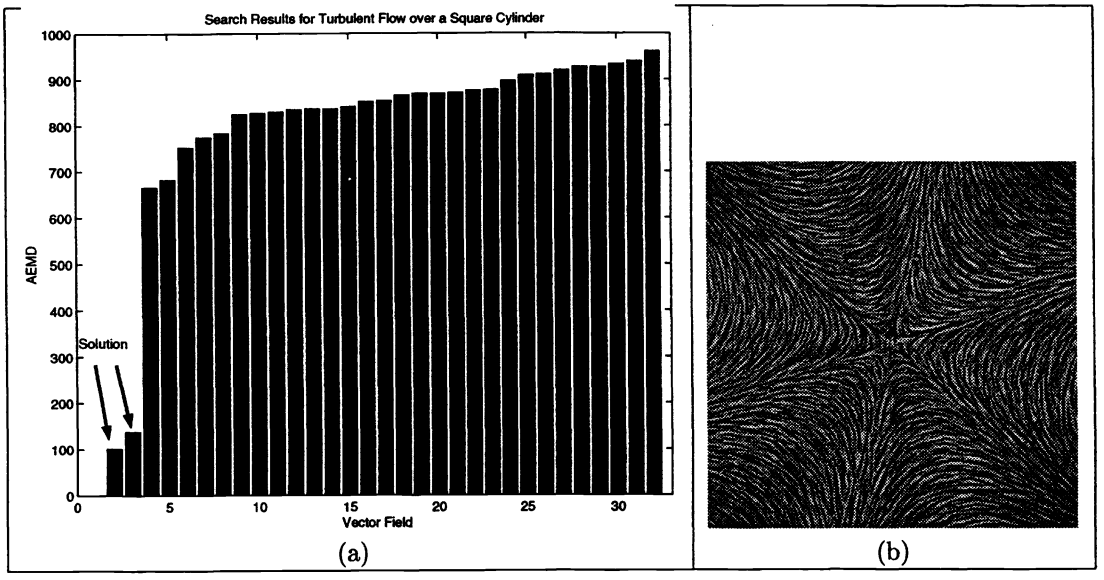

Figure 6. (a) Database search results sorted by AEMD. Bars marked solution with lowest energy (closest match) are in fact the fields computed in the same simulation as Figure 3(a) but a few times steps later. (b) Example of a higher order critical point, monkey saddle.

z_j, $j = 1, \ldots, n$ which are the zeros of $\vec{v}$ and the Poincaré index of $\vec{v}$ at z_j is the sum of the indices of the vector fields $\vec{w_k}$ at z_j.[32] Here the restriction is imposed that F_k is composed of linear factors. Therefore, a vector field can be described as the product of n linear factors:

$$\begin{aligned} \vec{v}(\vec{r}) &= E(z,\bar{z})e_1 \\ &= \prod_{k=1}^{n}(a_k z + b_k \bar{z} + c_k)e_1, \quad |a_k| \neq |b_k| \end{aligned} \tag{9}$$

where $z = x + iy$, $\bar{z} = x - iy$, e_1 is a basis matrix, and a_k, b_k, c_k are complex coefficients. A mapping of a_k, b_k to (α, β) can be formulated.[20] c_k merely positions the critical point. For example, the monkey saddle of Figure 6(b) at location z_j can be defined as $\vec{v}_{ms} = (z - z_j)^2 e_1$.

A difficulty before determining the decomposition of a higher order critical point is to detect and determine the order of the critical point within the data field. Scheuermann et al. begin to address this issue for second order critical points.[33] Winding numbers are calculated within the flow field, and if it is of index -2, a fit is used to determine the coefficients of the polynomial. Equation 9 is decomposed in a unique manner by normalizing the coefficient z (or $\bar{z}$ is $a_k = 0$). The coefficients of the decomposition can then be used to classify the components of the vector field in terms of (α, β). A similar methodology can be applied to even higher order critical points. Admittedly, one could argue that a higher order critical point could falsely be compared with a set of simple critical point separated by a distance. However with this definition, a higher order critical point can be construed as viewing the separated critical points from infinity.

6. CONCLUSIONS

Several new techniques have been presented for autonomous vector field comparisons. All these methods are based on the topological structure of a flow field. The simplest and fastest technique, EMD, uses just the critical points in the field. By further considering the connecting lines forming the separatrices of the vector field the quantitative measure between the fields is improved. Finally, a method is proposed to compare higher order critical points.

Future research is to incorporate higher order critical points such as attachment/detachment points within the comparison results and to analyze three dimensional flow fields with connectivity. Much work has been done in tensor topology[5] and comparison techniques in this area are under investigation.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

ACKNOWLEDGMENTS

The authors are most indebted to Yossi Rubner for providing the EMD code and Wim de Leeuw for providing the turbulent data set. Kerstin Kling is grateful to the Ernest-Solvay Foundation for the generous financial support during her stay at Stanford University.

REFERENCES

1. Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *Proc. IEEE Internations Conferences on Computer Vision*, 1998.
2. W.Barrett, "A survey of face recognition algorithms and testing results," in *Conference Record of the Thirty-First Asilomar Conference on Signals, Systems and Computers*, pp. 301–305, 1998.
3. H.-G. Pagendarm and F. H. Post, "Comparative visualization-approaches and examples," in *Visualization in Scientific Computing*, M. Göbel, H. Müller, and B. Urban, eds., pp. 95–108, Springer-Verlag, 1995.
4. Q. Shen, A. Pang, and S. Uselton, "Data level comparison of wind tunnel and computational fluid dynamics data," in *Proc. IEEE Visualization '97*, pp. 67–74, CS Press, Los Alamitos, CA., 1997.
5. T. Delmarcelle, *The Visualization of Second-Order Tensor Fields.* PhD thesis, Stanford University, 1994.
6. Y. Lavin, R. Batra, and L. Hesselink, "Feature comparisons of vector fields using earth mover's distance," in *Proc. IEEE/ACM Visualization '98*, pp. 413–415, North Carolina, October 1998.
7. W. E. Boyce and R. C. DiPrima, *Elemetrary Differential Equations and Boundary Value Problems*, John Wiley & Sons, fourth ed., 1986.
8. A. Perry and B. Fairlie, "Critical points in flow patterns," *Advances in Geophysics B* **18**, pp. 299–315, 1974.
9. U. Dallmann, "Topological structures of three-dimensional flow separation.," Tech. Rep. IB 221-82 A 07, DFVLR:Deutsche Forschungs-und Versuchsanstalt für Luft-un Ramumfahrt (German Research Institute for Aerospace), April 1983.
10. S. Rogers and D. Kwak, "An upwind differencing scheme for the time-accurate incompressible navier-stokes equations," in *Proceedings of the AIAA 6th Applied Aerodynamics Conference*, pp. 492–502, American Institute of Aeronautics and Astronautics, 1988.
11. Y. Rubner, L. J. Guibas, and C. Tomasi, "The earth mover's distance, multi-dimensional scaling, and color-based image retrieval," in *Proceedings of the ARPA Image Understanding Workshop*, 1997.
12. R. Batra, Y. Lavin, and L. Hesselink, "Topology based comparison technique for vector fields using earth mover's distance," in *Scientific Computing in Chemical Engineering II*, vol. 1, pp. 181–195, Springer Verlag, May 1998. Internation Workshop at Technical University Hamburg-Harburg at Hamburg, Germany. Submitted for publication.
13. S. T. Rachev, "The monge-kantorovich mass transference problem and its stochastic applications," *Theory of Probability and its Applications* **XXIX(4)**, pp. 647–676, 1984.
14. S. Smale, "On the average number of steps of the simplex method of linear programming," *Mathematical Programming* **27**, pp. 241–262, 1989.
15. Y. Rubner and C. Tomasi, "The earth mover's distance as a metric for image retrieval," Tech. Rep. STAN-CS-TN-98-86, Department of Computer Science, Stanford University, September 1998.
16. R. Verstappen and A. Veldman, "Spectro-consistent discretization of navier-stokes: a challenge to rans and les," *Journal of Engineering Mathematics* **34**, pp. 163–179, 1998.
17. B. T. Messmer and H. Bunke, "A network based approach to exact and inexact graph matching," Tech. Rep. IAM-93-021, University of Bern, September 1993.
18. M. R. Garey and D. S. Johnson, *Computers and intractability : a guide to the theory of NP-completeness*, A series of books in the mathematical sciences, W. H. Freeman, San Francisco, 1979.
19. R. Batra, K. Kling, and L. Hesselink, "Topology based vector field comparisons using graph methods," in *Late Breaking Hot Topics IEEE Visualization '99*, p. To be published, CS Press, Los Alamitos, CA., 1999.
20. K. Kling, "Topology based quantitative comparison between vector fields," Master's thesis, University of Hannover, 1999. Diplomarbeit.
21. B. Messmer, *Efficient Graph Matching Algorithms for Preprocessed Model Graphs.* PhD thesis, University of Bern, 1996.
22. A. K. Wong, M. You, and S. Chan, "An algorithm for graph optimal monomorphism," *IEEE Transactions on Systems, Man, and Cybernetics* **20**(3), pp. 628–636, 1990.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use

23. S. Russell and P. Norvig, *Artificial intelligence : a modern approach*, Prentice Hall, Englewood Cliffs, N.J, 1995.
24. J. Kittler, W. Christmas, and M. Petrou, "Probabilistic relaxation for matching of symbolic structures," in *Workshop on Structural and Syntactic Pattern Recognition*, pp. 471–480, (Bern, Switzerland), 1992.
25. S. Mehta and L. Fulop, "A neural algorithm to solve the graph matching problem," *Proc. of Int. Neural Network Conference INNC* **1**, pp. 262–266, 1990.
26. L. Xu and E. Oja, "Improved simulated annealing, boltzmann machine and attributed graph matching," *Lecture Notes in Computer Science 412* **1**, pp. 151–161, 1990.
27. S. Umeyama, "An eigendecomposition approach to weighted graph matching problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI* **10(5)**, pp. 695–703, 1988.
28. H. Almohamad and S. Duffuaa, "A linear programming approach fo the weighted graph matching problem," *PAMI* **15**, pp. 522–525, May 1993.
29. R. Batra and L. Hesselink, "Feature comparisons of 3-d vector fields using earth mover's distance," in *Proc. IEEE Visualization '99*, p. To be published, IEEE Computer Society Press, Los Alamitos, CA, 1999.
30. G.Scheuermann, H.Krüger, M.Menzel, and A.Rockwood, "Visualizing non-linear vector field topology," *Computer Graphics and Its Applications, Special Edition* , 1998.
31. G.Scheuermann, H.Krüger, M.Menzel, and A.Rockwood, "Visualization of higher order singularities in vector fields," in *Proc. IEEE Visualization '97*, pp. 67–74, CS Press, Los Alamitos, CA., 1997.
32. G.Scheuermann, H.Hagen, and H.Krüger, "An interesting class of polynomial vector fields," in *Mathematical Methods for Curves and Surfaces II*, L. M.Daehlen, T.Lyche, ed., pp. 429–436, Vanderbilt University Press, 1998.
33. G.Scheuermann, H.Kruger, and A.Rockwood, "Visualizing non-linear vector field topology," *IEEE Transactions on Visualization and Computer Graphics* **4**(2), pp. 109–116, 1998.

Downloaded From: https://www.spiedigitallibrary.org/conference-proceedings-of-spie on 10 Apr 2019
Terms of Use: https://www.spiedigitallibrary.org/terms-of-use