

Eventbasierte Visualisierung zeitabhängiger Datensätze

Florian Weidner

Technische Universität Dresden

Fakultät Informatik

Professur für Computergraphik und Visualisierung

Helmholtzstraße 10, 01069 Dresden

Email: Florian.Weidner@tu-dresden.de

Zusammenfassung—Die heutige Wissenschaft erzeugt bei Experimenten enorme Datenmengen. Der Large-Hadron-Collider des CERN¹ produziert pro Jahr 15 Petabytes an Messergebnissen². Diese Daten müssen analysiert und bearbeitet werden. Ein visuelles Analysieren der Rohdaten durch den Menschen ist aufgrund dieser Datenmenge nicht mehr möglich. Neue Methoden der Analyse müssen angewandt werden um mit den enormen Mengen umzugehen. Der hier vorgestellte Ansatz der eventbasierten Visualisierung zeitabhängiger Datensätze will dieses Problem lösen. Durch eine zweifache Datenreduktion wird die Menge an Werten reduziert und eine visuelle Analyse durch den Menschen vereinfacht. Der vorliegende Artikel stellt die eventbasierte Visualisierung zeitabhängiger Datensätze vor und erläutert beispielhaft Techniken die einerseits die Datenreduktion und andererseits die Visualisierung der neu berechneten Daten betreffen. Dieser Ansatz kann die Datenanalyse vereinfachen und schneller gestalten. Die richtige Visualisierungstechnik ist hierfür ebenso entscheidend wie eine gut abgestimmte Datenreduktion.

I. EINFÜHRUNG

Wissenschaftliche Forschungsprojekte produzieren oft, vor allem im naturwissenschaftlich-technischen Bereich, eine große Menge an Messwerten. Diese Daten müssen analysiert werden. In dem Artikel von Laney et al. [1] wird das Verhalten von zwei Flüssigkeiten die sich mischen (Rayleigh-Taylor-Instabilität) mit einer Kamera aufgenommen und anschließend analysiert. Hierfür müssen die Kamerabilder paarweise verglichen werden. Der Mensch kann dies aufgrund der vielen Bilder bzw. Daten nicht bewerkstelligen.

Die eventbasierte Visualisierung zeitabhängiger Datensätze versucht einerseits durch Datenreduktion und andererseits durch das Berechnen neuer Daten aus den Rohdaten, den Analyseprozess zu verbessern und zu erleichtern. Hauptbestandteile hierbei sind Features und Events. Features, oder auch Merkmale, sind charakteristische Repräsentanten im Datensatz, die sich über die Zeit verändern. Die Veränderungen dieser Features nennt man Events oder Ereignisse. Die Definition der Features als auch der Events ist beliebig und abhängig von der Domäne, in der das Verfahren eingesetzt wird. Als Resultat erhält man einen neuen Datensatz, bestehend aus eben diesen Features und Events.

Trotz der Datenreduktion kann die Datenmenge immer noch groß sein. Zur Analyse sind spezielle Visualisierungstechniken

erforderlich. Hierfür ist eine wohlüberlegte und gut gestaltete Anwendung erforderlich. Sie muss ausreichend Möglichkeiten zum Darstellen der Rohdaten als auch der resultierenden Daten bieten. Es existiert eine Fülle von Anzeigetechniken die hierfür in Frage kommen. Jede hat Vor- und Nachteile die es abzuwägen gilt.

Der vorliegende Artikel ist wie folgt gegliedert: Der folgende Abschnitt definiert Features und Events im Kontext der eventbasierten Visualisierung. Abschnitt III beschäftigt sich mit den Möglichkeiten, Features aus einem Datensatz zu extrahieren. Vom Feature zum Event (Abschnitt IV) beschreibt das Verfahren der Eventextraktion. Der vorletzte Teil dieses Artikels (Abschnitt V) beschäftigt sich mit den verschiedenen Möglichkeiten Features und Events zu visualisieren. Abschließend wird der Artikel zusammengefasst und ein Ausblick gegeben.

II. FEATURES UND EVENTS

Eingangs genanntes Beispiel der beiden sich mischenden Flüssigkeiten eignet sich, um Features zu erklären. Zwei Flüssigkeiten unterschiedlicher Dichte sind in einem Behälter. Durch eine Druckwelle werden sie vermischt. Es entstehen Verwirbelungen oder Blasen. Abbildung 1 zeigt vier Momentaufnahmen von zwei Flüssigkeiten in unterschiedlichen Zoomstufen. Die Turbulenzen werden stärker und die Strukturen komplizierter. Eine visuelle Analyse der Rohdaten durch den Menschen ist aufgrund der hohen Komplexität nicht mehr möglich. Mit der Einführung von Features wird diesem Problem entgegen getreten.

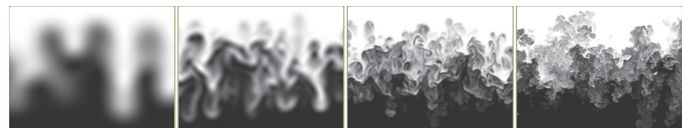


Abbildung 1: Vier Momentaufnahmen zweier sich mischender Flüssigkeiten in unterschiedlichen Zoomstufen [1]

A. Feature

Ein Feature ist eine vom Wissenschaftler festgelegte Beschreibung eines charakteristischen Repräsentanten im Datensatz. Weiterhin kann jedes Feature Attribute haben. Im Beispiel der beiden Flüssigkeiten könnte das Feature die Blase und ein

¹www.cern.ch

²www.lhc-facts.ch

Attribut das Volumen dieser sein. So muss nicht mehr der komplette Datensatz analysiert werden, sondern nur die Features. Durch das Abstrahieren der Rohdaten erreicht man eine Informationsreduktion. Es ergibt sich aber auch der Vorteil, dass in der Visualisierung bzw. der Analyse nur die Daten behandelt werden, die von Interesse sind. In diesem Fall, das Verhalten der Blasen.

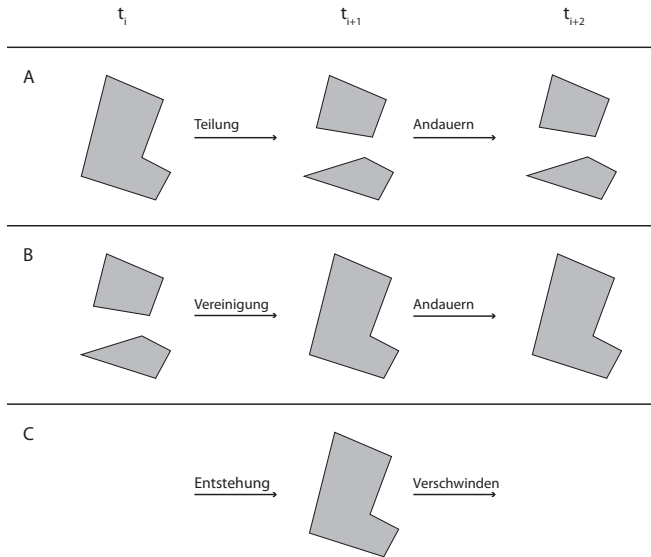


Abbildung 2: Mögliche Definitionen von Events

B. Event

Ein Event beschreibt das Übergangsverhalten von einem oder mehreren Features aus einem Zeitschritt t_i hin zum Zeitschritt t_{i+1} . Events dienen dazu, die Entwicklung von Features darzustellen. Abbildung 2 zeigt mögliche Eventdefinitionen. Im Beispiel der sich mischenden Flüssigkeiten könnten die Polygone aus der Abbildung für Blasen stehen. Teilt sich eine Blase: *Teilung*. Vereinigen sich zwei Blasen: *Vereinigung*. Entsteht eine neue Blase: *Entstehung*. Analog dazu existiert das *Verschwinden*. Ändert sich ein Feature von einem Zeitschritt zum nächsten nicht, kann dies unter dem Namen *Andauern* als Event definiert werden. Analoge Featuredefinitionen wurde von Samtaney et al. [2] und Reinders et al. [3] gewählt.

III. FEATUREEXTRAKTION

Featureextraktion beschreibt zwei Vorgänge: einerseits die Definition der Features, andererseits deren Extraktion aus den Rohdaten. Die Definition von Featurebeschreibungen wird häufig von Fachpersonal begleitet, da sie Repräsentanten identifizieren und deklarieren können. Die Extraktion dieser Features kann durch verschiedene Techniken geschehen. Als Beispiele können Segmentierungsalgorithmen, Isoflächenextraktion und die Extraktion von Morse-Smale Komplexen genannt werden. Die letzten beiden Techniken werden im Folgenden kurz beschrieben.

A. Isoflächen

Wenn Isoflächen als Feature verwendet werden, werden Punkte mit gleichen oder ähnlichen Werten wie z. B. Temperatur oder

Position zu einer geschlossenen Region verbunden (z. B. mit dem Marching-Cubes Verfahren). Die so berechneten Regionen können als Features verwendet werden. Die Entwicklung dieser Regionen wird dann zur Eventdetektion genutzt. Als mögliche Attribute kann der Flächeninhalt oder der Schwerpunkt der Isofläche herangezogen werden.

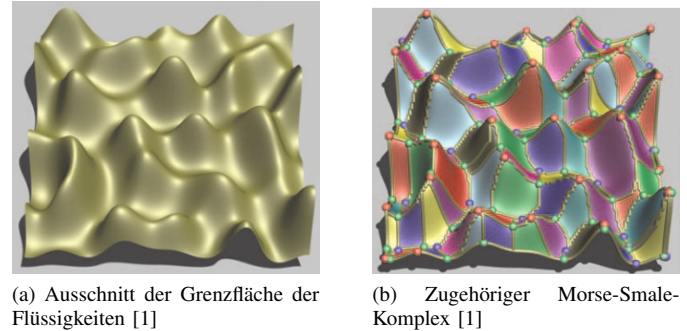


Abbildung 3: Grenzflächenausschnitt aus den Rohdaten (a) mit dem zugehörigen Morse-Smale-Komplex (b)

B. Morse-Smale-Komplex

Als zweites Beispiel einer Featuredefinition wird im Folgenden der Morse-Smale-Komplex (MSK) erläutert. Aufgrund der Komplexität wird nur schematisch auf das Vorgehen eingegangen. Grundlegende Informationen zur Morse-Theorie sind in [4] zu finden.

Der MSK beschreibt die Topologie einer Oberfläche mathematisch. Eine mögliche Anwendung sieht wie folgt aus: Zuerst wird eine Fläche anhand eines Koordinatensystems in Zellen unterteilt. An jedem Punkt im Gitter wird der Gradient berechnet. Man erhält ein Gradientenfeld. Mithilfe dieses Gradientenfeldes können Maxima, Minima und Sattelpunkte extrahiert werden. Diese Punkte teilen die Gesamtfläche in Regionen ein. Das Ergebnis dieser Berechnungen ist der Morse-Smale-Komplex.

Diese Technik wird von Laney et al. [1] dazu verwendet, die Feature-Extraktion im bereits genannten Beispiel der sich mischenden Flüssigkeiten zu realisieren. Abbildung 3a zeigt einen Ausschnitt aus der Grenzfläche zweier Flüssigkeiten. Die Erhöhungen und Vertiefungen verdeutlichen die sich mischenden Flüssigkeiten. Die kritischen Punkte (Minima, Maxima und Sattelpunkte) sind in Abbildung 3b veranschaulicht. Statt eines kompletten pixel-basierten Vergleichs der Grenzfläche von t_i zu t_{i+1} können jetzt diese markanten Punkte berechnet und in der Eventextraktion verwendet werden.

IV. VOM FEATURE ZUM EVENT

Die extrahierten Features müssen verknüpft werden, um Events zu berechnen. Hierzu existieren zwei mögliche Ansätze:

- 1) One-Pass: Frame-To-Frame
- 2) Two-Pass

Die beiden Ansätze unterscheiden sich in der Ausführungsreihenfolge der Schritte *Extraktion* und *Vergleichen*. Beim One-Pass-Verfahren werden die Features

der Frames t_i und t_{i+1} extrahiert und dann miteinander verglichen. Danach werden die Features von Frame t_{i+2} extrahiert und mit t_{i+1} verglichen. Dieses Vorgehen wird bei allen existierenden Frames angewandt. Der Vorteil dieser Technik ist, dass die Berechnung von Events an jeder Stelle des Datensatzes beginnen kann. Bei einer Änderung der Featuredefinition kann schnell mit den Daten gearbeitet werden.

Ein entscheidender Nachteil ist, dass hier für jedes Paar von Frames das hochkomplexe Korrespondenzproblem zu lösen ist. Das Korrespondenzproblem besagt, dass ein Feature aus Frame t_i mit allen Features aus Frame t_{i+1} verglichen werden muss, um den bestmöglichen Korrespondenten zu finden. Dies kann bei einer großen Featureanzahl enorm rechenintensiv werden. Um dieses Problem möglichst schnell zu lösen, sind häufig Heuristiken notwendig. Zudem können Verdeckungen und Mehrdeutigkeiten schwer oder nur unter der Anwendung von weiteren Techniken erkannt und aufgelöst werden.

Eine solche Technik ist das Vorhersagen von potenziellen Features. Wenn die Position bzw. die Region eines Features aus Frame t_i im Frame t_{i+1} vorhergesagt werden kann, muss das Feature nicht mit allen potenziellen Features verglichen werden, sondern nur mit denen, die in der vorhergesagten Region liegen. Zu beachten ist, dass dieses Vorgehen zu Fehlern führen kann. Wenn bspw. große Veränderungen in der Position vorliegen, wird ein falsches Event erkannt.

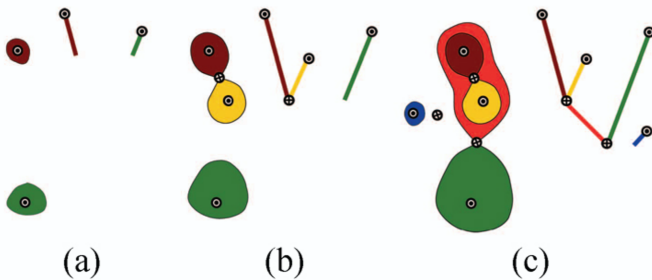


Abbildung 4: Isoflächen mit Merge-Tree [5]

Das Two-Pass-Verfahren extrahiert im ersten Schritt alle Features von allen Frames und vergleicht dann die jetzt komplett zur Verfügung stehenden Features. Dies hat zum Vorteil, dass die Events einfacher erkannt werden, da nicht nur Features aus zwei Frames genutzt werden können, sondern aus mehreren. Die verschiedenen Charakteristiken sind so einfacher zu erkennen. Eine mögliche Umsetzung dieser Technik sind die Merge-Trees in Verbindung mit Meta-Graphen (umgesetzt von Widanagamaachchi et al. [5]).

Ein Merge-Tree repräsentiert die Features eines Frames in einer Baumstruktur. Ein Knoten repräsentiert ein Feature. Eine Verbindung zwischen Knoten repräsentiert eine *enthält*-Relation.

Um den Merge-Tree zu erklären, bietet es sich an, Isoflächen als Features zu benutzen: Jedes Mal wenn eine Isofläche gebildet wird, wird ein Knoten im Merge-Tree erstellt. Wenn eine Fläche B eine Fläche A einschließt, werden die Knoten A und B verbunden (Relation: Fläche B enthält Fläche A). Diese

Technik wird auf alle Isoflächen eines Frames angewandt (vgl. Abb. 4). Ebenso wird für jeden Frame ein Merge-Tree erstellt.

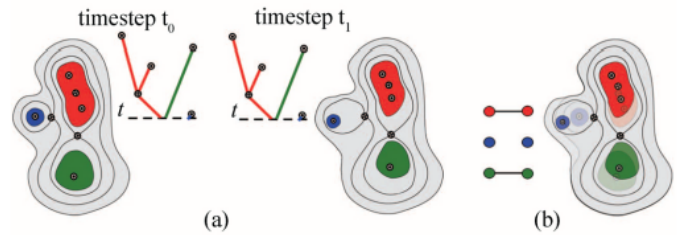


Abbildung 5: Meta-Graph [5]

Um Events zu erhalten, werden die Merge-Trees der Frames verglichen und die Ergebnisse in einem sogenannten Meta-Graphen gespeichert (vgl. Abb. 5). Der Meta-Graph enthält Informationen über die Entwicklung der in den Merge-Trees enthaltenen Features. Knoten repräsentieren Events. Verbindungen zwischen Knoten repräsentieren Events. Beim Bilden eines Meta-Graphen wird wie folgt vorgegangen: Zwei Merge-Trees werden miteinander verglichen. Wenn zwei Features übereinstimmen, wird im Meta-Graph (vgl. 5b) eine Verbindung für diese Features erstellt. Es muss jedoch keine vollständige Übereinstimmung vorhanden sein. Eine teilweise Übereinstimmung oder eine Nachbarschaft in einem gewissen Bereich kann ebenfalls dazu führen, dass eine Verbindung zwischen zwei Knoten im Meta-Graphen erstellt wird. In einer Visualisierung kann dann der Meta-Graph traversiert werden. Er stellt die direkte Repräsentation von Events dar. Durch die Parameterbereiche, die bei der Erstellung des Meta-Graphen genutzt werden, ist zusätzlich eine nachträgliche schnelle Anpassung der Eventdefinitionen ohne komplette Neuberechnung möglich.

Das One-Pass Verfahren bietet zwar initial einen schnelleren Zugriff auf die Eventdaten an einem ausgewählten Bereich des Datensatzes, benötigt allerdings eine erheblich höhere Rechenleistung, die aus der Lösung des Korrespondenzproblems resultiert. Zudem kann hier nicht immer sichergestellt werden, dass Verdeckungen und Mehrdeutigkeiten richtig aufgelöst werden. Das Two-Pass Verfahren benötigt eine längere Zeit um erste Ergebnisse nutzen zu können, da zuerst alle Features berechnet werden. Durch die eingesetzten Parameterbereiche beim Meta-Graphen muss im Gegensatz zum One-Pass Verfahren keine komplette Neuberechnung der Events stattfinden. Es muss lediglich der Meta-Graph angepasst werden. Zudem kann das Korrespondenzproblem umgangen werden.

V. VISUALISIERUNG

Nachdem Events extrahiert wurden, muss eine passende Visualisierung gefunden werden. Diese sollte eine gute Vergleichbarkeit der Daten, sowohl in der Zeitdimension als auch bzgl. der Attribute, bieten. Um den Datensatz einschätzen und in ihm effizient navigieren zu können, ist ein Gesamtüberblick notwendig. Eine dynamische Ansicht mit veränderbaren Parametern wie z. B. Attributdefinitionen, Eventdefinitionen o.ä. sind wünschenswert. Ebenso relevant ist die Darstellung des Eventtyps als Kern der hier vorgestellten Technik.

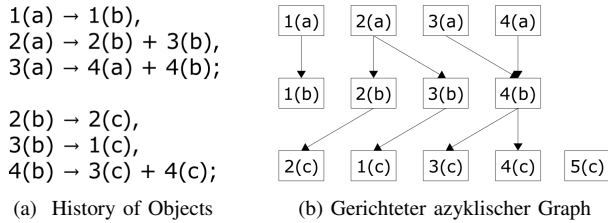


Abbildung 6: Events in zwei Darstellungen: Gerichteter azyklischer Graph und History of Objects

A. History of Objects und Gerichtete azyklische Graphen

Die Darstellung aus Abbildung 6a, welche als *History of Objects* bezeichnet und in [2] genannt wird, setzt auf eine formelartige Darstellung der Events. Sie eignet sich, um ein Feature bzw. eine kleine Gruppe von Features zu beschreiben und deren Entwicklung zu verfolgen. Mit steigender Menge an Features, und damit Events, wird diese Art der Visualisierung unübersichtlich. Als Weitere, verwandte Visualisierung nennen Samtaney et al. in [2] *gerichtete azyklische Graphen*. Sie sind ebenfalls gut für Ausschnitte geeignet (Abb. 6b). Einzelne Phänomene im Datensatz können übersichtlich, strukturiert und klar dargestellt werden. Bei beiden Darstellungen können statische Sachverhalte gut verdeutlicht werden. Mit größer werdender Datenmenge wird die Visualisierung immer unangemessener.

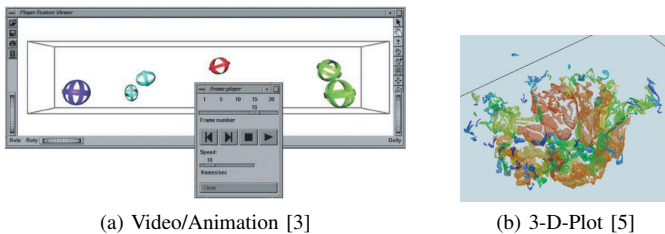


Abbildung 7: Videoplayer für Events und 3D-Plot

B. Video/Animation

Mit *Videos* bzw. *Animationen* (vgl. Abb. 7a) kann der Ablauf im Allgemeinen gut verdeutlicht werden. Im Beispiel der zwei sich mischenden Flüssigkeiten kann hier der Mischvorgang abstrahiert und mit Events anschaulich dargestellt werden. Bei einem Player mit verschiedenen Funktionen wie Play, Pause, Zeitlupe usw. wird diese Technik weiter verbessert. Schwer fällt das Vergleichen von nicht unmittelbar benachbarten Zeitpunkten als auch von Attributwerten. Kurzlebige Events lassen sich schwer erkennen.

C. 2-D/3-D Plot

Ein statischer *Plot* visualisiert Daten in einem Koordinatensystem an einem gewählten Zeitpunkt und kann zur Verdeutlichung von Attributwerten wie z. B. des Grenzwertes einer Isofläche oder der Position einer Blase verwendet werden. In Abbildung 7b ist einerseits durch die farbliche Codierung und andererseits durch die Position im 3-D-Raum eine quantitative

Analyse der Daten zu diesem Zeitpunkt möglich. Ein Vergleich von mehreren Frames ist ohne Weiteres nur schwer möglich. Da ein Plot nur die Darstellung eines einzelnen Frames ist, bleibt der Eventtyp verborgen.

D. Eventgraph

Eventgraphen sind 2-D-Darstellungen von Features und deren Entwicklung. In x-Richtung werden bspw. die Daten über die Zeit und in y-Richtung über eine Messgröße (z. B. die Anzahl der Blasen in zwei sich mischenden Flüssigkeiten) aufgetragen. Der Eventgraph bietet, im Gegensatz zu den vorangegangenen Visualisierungen, eine Vergleichbarkeit von Attributen und weit auseinanderliegenden Frames.

Bei großen bzw. dichten Datenmengen wie in Abb. 8 ist ein detailliertes Vergleichen schwer möglich. Wenn der Eventgraph eine Zoom-Funktion anbietet, kann jedoch ein Gesamtüberblick vorhanden sein und auch im Detail analysiert werden.

Eine Verbesserung, die bei Eventgraphen angewandt werden kann, ist die Nutzung von Greedy- und optimierten Layouts. Zu Beginn der Berechnung werden alle Features mit ihren Verbindungen nicht optimiert in den Graphen eingetragen (Greedy-Layout), um schnell Daten anzuzeigen. Zu einem späteren Zeitpunkt kann die Darstellung optimiert werden, um zum Beispiel Überschneidungen zu minimieren und somit die Verfolgbarkeit von Features besser zu gestalten (optimiertes Layout).

In Abbildung 8 werden die berechneten Features in einem solchen Eventgraphen dargestellt. Hier kann das Verhalten der Blasen gut verfolgt werden. Zu beachten ist, dass hier keine Skala an der y-Achse angelegt ist. Da die y-Achse zufällig sortiert ist, könnte die Darstellung optimiert werden, um Überschneidungen zu eliminieren.

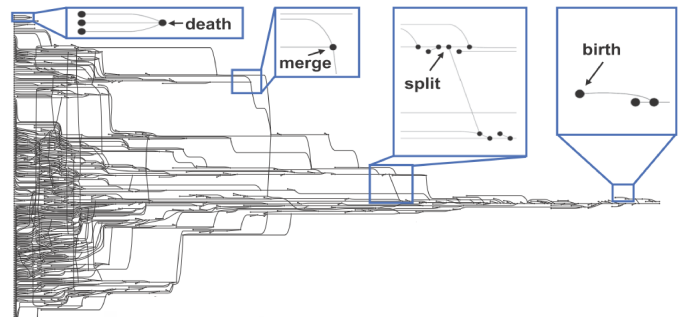


Abbildung 8: Event-Graph [1]

E. Korrelierte Visualisierung

Wenn Visualisierungen kombiniert werden, fasst man diese unter dem Begriff *Korrelierte Visualisierung* zusammen. Eine mögliche Komposition von Anzeigetechniken wäre der Eventgraph, ein Videoplayer sowie ein 3-D-Plot. Wird im Videoplayer die Entwicklung der Features abgespielt, aktualisiert sich parallel dazu der 3-D-Plot, welcher die Attributwerte anzeigt und im Eventgraphen deutet eine senkrecht zur x-Achse gezogene Linie den aktuellen Zeitpunkt an und erlaubt schnell

	Vergleich (Attribute)	Vergleich (temporal)	Gesamtüberblick	Dynamische Ansicht	Eventtyp
History of Objects	-	-	-	-	+
DAG	-	+	-	-	+
Video/Animation	-	-	-	+	+
2-D/ 3-D-Plot	+	-	+	-	-
Eventgraph	-	+	+	-	+
Korrelierte Ansichten	+	+	+	+	+

Tabelle I: Vergleich der Visualisierungstechniken nach eigenen Kriterien

einen temporalen Vergleich. Diese verschiedenen Visualisierungen können je nach Anwendungsfall kombiniert werden und dem Nutzer so ermöglichen, die Events zu verfolgen und den Datensatz gut zu analysieren.

VI. VERWANDTE ARBEITEN

Reinders et al. [3] beschreiben in ihrem Artikel das grundlegende Vorgehen beim Extrahieren von Features und Events mit mathematischen Definitionen. Weiterhin werden grundlegende Visualisierungstechniken genannt. In der frühen Arbeit von Samtaney et al. [2] beschreiben die Autoren ebenfalls mathematische Grundlagen zur Analyse von Features und Events.

VII. ZUSAMMENFASSUNG

Es wurde gezeigt, dass die eventbasierte Visualisierung zeitabhängiger Datensätze eine Methode ist, die den Umgang mit großen Daten beschleunigt und Zusammenhänge zwischen einzelnen Frames analysiert. Durch die Extraktion von Features wird der zu bearbeitende Datensatz im Vergleich zu den Rohdaten verkleinert. Möglichkeiten hierzu sind unter anderem Isoflächen oder Morse-Smale-Komplexe. Zur Eventextraktion existieren zwei grundlegende Ansätze: One-Pass und Two-Pass. One-Pass zeigt Daten schneller an, muss aber mit dem Korrespondenzproblem umgehen und kann fehlerhaft sein. Deswegen sollte Two-Pass verwendet werden. Es braucht zwar initial länger um die Visualisierung aufzubauen, kann aber das Korrespondenzproblem umgehen.

Die verschiedenen vorgestellten Anzeigetechniken haben alle Stärken und Schwächen. Eine korrelierte Ansicht verschiedener Techniken ermöglicht eine gute Arbeit mit dem Datensatz und vereint viele Möglichkeiten der Analyse (vgl. Tabelle I). Es können Attribute in einem als auch von verschiedenen Frames verglichen werden. Zudem hat man eine dynamische Ansicht und einen Gesamtüberblick. Der Eventtyp kann ebenfalls sichtbar gemacht werden. Diese Art der Visualisierung ist Einzelvisualisierungen vorzuziehen.

Denkbare Erweiterungen der eventbasierten Visualisierung zeitabhängiger Datensätze betreffen einerseits die Schritte Definition und Extraktion und andererseits die Visualisierung. Bei Ersterem sind noch schnellere Algorithmen nützlich, um Datensätze schnell zu analysieren (sowohl bei der Feature- als auch bei der Eventextraktion). Eine Anwendung, die kritische Frames oder Bereiche (z. B. solche mit hoher Dynamik oder einer bestimmten Anzahl eines Eventtyps) vorauswählt und dem Nutzer zu Beginn anbietet, würde die Analyse unterstützen und zur weiteren Verbesserung beitragen.

LITERATUR

- [1] D. Laney, P. T. Bremer, a. Mascarenhas, P. Miller, and V. Pascucci, "Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities," *IEEE transactions on visualization and computer graphics*, vol. 12, no. 5, pp. 1053–60, 2006. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/17080834>
- [2] R. Samtaney, D. Silver, N. Zabusky, and J. Cao, "Visualizing features and tracking their evolution," *Computer*, vol. 27, no. 7, pp. 20–27, Jul. 1994. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=299407>
- [3] F. Reinders, F. H. Post, and H. J. Spoelder, "Visualization of time-dependent data with feature tracking and event detection," *The Visual Computer*, vol. 17, no. 1, pp. 55–71, Feb. 2001. [Online]. Available: <http://link.springer.com/10.1007/PL00013399>
- [4] J. W. Milnor, "Morse theory," *Princeton university press*, vol. No. 51, 1963.
- [5] W. Widanagamaachchi, "Interactive exploration of large-scale time-varying data using dynamic tracking graphs," *Large Data Analysis and Visualization*, pp. 9–17, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6378962