LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
INSTITUT FÜR INFORMATIK
LEHRSTUHL FÜR MOBILE UND VERTEILTE SYSTEME

MASTERARBEIT

Der Vergleich von Trajektorien

Wie man mit Pfaden von bewegten Objekten umgeht

Philipp Benedikt Moers

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN
INSTITUT FÜR INFORMATIK
LEHRSTUHL FÜR MOBILE UND VERTEILTE SYSTEME

MASTERARBEIT

Der Vergleich von Trajektorien

Wie man mit Pfaden von bewegten Objekten umgeht

Philipp Benedikt Moers

Aufgabensteller:	Prof. Dr. Claudia Linnhoff-Popien
Betreuer:	Dr. Martin Werner
Abgabetermin:	9. Dezember 2016

Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 9. Dezember 2016

.....

(Unterschrift des Kandidaten)

Zusammenfassung

Die Anwendungen für spatiotemporale Trajektorien – Beschreibungen von Pfaden bewegter Objekte durch den Raum als Funktion der Zeit – sind so zahlreich wie verschieden und reichen von der Nutzung der Standortdaten von Smartphones für Location-Based-Social-Networks über Handschrifterkennung bis hin zur automatisierten Erkennung von Anomalien in der Videoüberwachung. Eines der Kernprobleme dabei ist der Vergleich von Trajektorien. So wie die Anwendungen sind auch die verwendeten Techniken zur Quantifizierung der Ähnlichkeit äußerst divers. Der Fachbereich hat eine Fülle von Ähnlichkeitsmaßen hervorgebracht, wobei deren Eigenschaften und die damit einhergehenden Vor- oder Nachteile für eine konkrete Anwendung häufig nicht offensichtlich sind.

Diese Arbeit präsentiert zunächst eine umfassende Übersicht über existierende Vergleichstechniken und berücksichtigt dabei sowohl sehr verbreitete als auch recht außergewöhnliche Ansätze. Darüber hinaus leitet sie aus inhärenten Charakteristika von Ähnlichkeitsmaßen Klassen ab, die ihre objektive Beurteilung erlauben, und ordnet sie schließlich in die definierten Klassen ein. Das Ergebnis ist ein Klassensystem, mit dem sämtliche Techniken für den Vergleich von Trajektorien ihrerseits systematisch miteinander verglichen werden können.

Inhaltsverzeichnis

1 Gegenstand der Arbeit

Seien es Menschen mit Smartphones oder öffentliche Verkehrsmittel mit Positionsbestimmungssystemen, sich virtuell bewegende Objekte wie Avatare in Computerspielen oder tatsächlich fliegende Skispringer: Eine Folge der anhaltenden Digitalisierung ist es, dass wir Pfade von bewegten Objekten aufnehmen, speichern, verarbeiten und daraus Informationen gewinnen können. Diese Pfade durch den Raum als Funktion der Zeit heißen Trajektorien und bilden die grundlegendste Form von spatiotemporalen Daten. Aus technologischen, sozialen und kommerziellen Gründen nimmt die Menge solcher Daten rapide zu [DN05]. In Anbetracht dieser Entwicklung stellt sich die Frage nach einem sinnvollen Umgang mit Trajektorien. Dabei ist die Suche nach ähnlichen Trajektorien eines der Kernprobleme:

> „One of the most important requirements for analyzing trajectories is to search for objects with similar trajectories and cluster them." [HKL05, Kap. 1]

Diese Tatsache ist im Fachgebiet gut bekannt [MdB02, YAS03, DTS+08, WMD+13, YAS03]. Um ähnliche Trajektorien zu finden, muss man sie miteinander vergleichen und ihre (Un-)ähnlichkeit zu diesem Zweck durch ein sogenanntes Ähnlichkeitsmaß (engl. *similarity measure*) quantifizieren. Der Entwurf eines solchen Ähnlichkeitsmaßes ist der Dreh- und Angelpunkt insbesondere bei der Clusteranalyse [PF06, Abschn. 2]. Infolgedessen gibt es sehr viel Literatur, die sich mit dieser Thematik beschäftigt und Techniken für den Vergleich von Trajektorien vorschlägt [DTS+08]. Trotzdem beklagen manche Wissenschaftszweige den Mangel an solchen Methoden [RMJ07, Kap. 1]. Grund dafür ist möglicherweise unter anderem, dass es sich um ein recht junges Forschungsgebiet handelt [FGT07, Kap. 1].

Die Fülle an Ähnlichkeitsmaßen und Methoden für den Vergleich von Trajektorien bei gleichzeitiger Intransparenz über deren Eigenschaften und die daraus resultierenden Vor- und Nachteile für eine Anwendung machen es schwierig, sich einen Überblick über sie zu verschaffen oder ihn zu bewahren, insbesondere als Anfänger in dem Forschungsgebiet [WMD+13, Kap. 1]. Bisher ermangelt es an Möglichkeiten, solche Vergleichstechniken in konsistenter Form kennenzulernen, nachzuschlagen, und deren Zusammenhänge zu verstehen, um sie einordnen und gegeneinander abwägen zu können. Auch bei neuen

Ähnlichkeitsmaßen, die ihre eigene Überlegenheit postulieren, fällt es oft schwer, sie in Relation zu existierenden Techniken zu setzen. Die vorliegende Arbeit bedient genau diesen Bedarf.

Die Arbeit gliedert sich wie folgt: Zunächst werden wir die Vielfältigkeit der Anwendungen von Vergleichstechniken auf Trajektorien und die damit verbundene Relevanz der Thematik durch diverse Anwendungsbeispiele unterstreichen und so unsere Arbeit motivieren (Kapitel 2). Anschließend betrachten wir verwandte Arbeiten, grenzen darauf aufbauend die bestehende Problematik ein und erläutern die Vorgehensweise, uns mit dieser Problematik zu befassen (Kapitel 3). Der materielle Teil beginnt mit einer systematischen Einführung in die verwendete Terminologie und wichtige Techniken im Zusammenhang mit dem Umgang mit Trajektorien (Kapitel 4). Daran schließt sich der Hauptteil der Arbeit an, der im Wesentlichen aus zwei Komponenten besteht: In der ersten (Kapitel 5) findet sich eine wohlerwogene Auswahl und Erklärung von Vergleichstechniken für Trajektorien in Form von Ähnlichkeitsmaßen. In der zweiten (Kapitel 6) werden Kriterien in Form von Klassifikationen aufgestellt, nach denen sich Ähnlichkeitsmaße ihrerseits vergleichen lassen. Kapitel 7 fasst die Arbeit letztendlich zusammen und ordnet sie in den wissenschaftlichen Kontext ein.

2 Anwendungen

Dieses Kapitel soll einen Eindruck davon verschaffen, wie weitrechend und unterschiedlich die Anwendungen des Trajektorienvergleiches sind. Es dient auf diese Weise als Motivation für die genauere Untersuchung der dafür verwendeten Techniken.

2.1 Automatisierte Videoüberwachung

Durch Videoüberwachung fallen große Mengen an Daten an, deren manuelle Auswertung nicht nur mühsam, sondern ab einer gewissen Größenordnung auch nicht mehr handhabbar ist. Es bedarf daher automatisierter Methoden, bewegte Objekte in solchen Videos zu erkennen, zu verfolgen und sinnvoll zu verarbeiten. Dies ist Gegenstand der Forschung [CM99, HBC+05, BER+03].

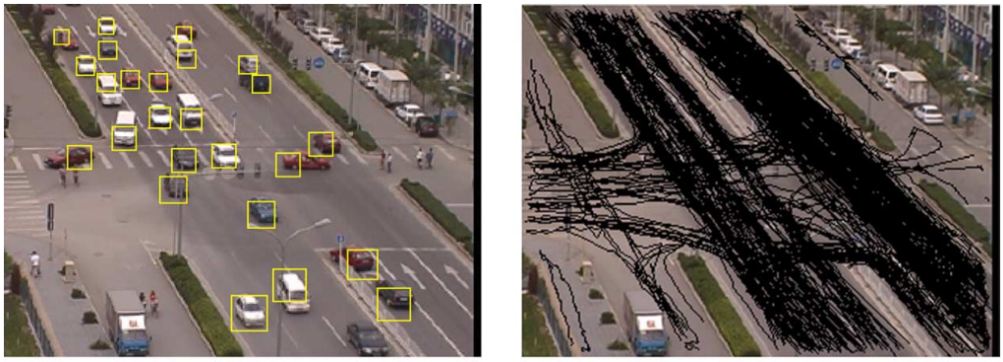


Abbildung 2.1: Tracking von bewegten Objekten und erzeugte Trajektorien [HXF+07].

Durch die Verfolgung (engl. *tracking*) von Fahrzeugen, Fußgängern oder anderen bewegten Objekten entstehen also Trajektorien. Möchte man Anomalien in den Bewegungen feststellen, muss man diese Menge von Trajektorien vergleichen. Das kann zum Beispiel sinnvoll sein, weil ein Einbrecher vor einer Überwachungskamera einen anderen Weg zu einer anderen Zeit zurücklegt als eine sich rechtmäßig in einer sicheren Umgebung aufhaltende Person. Wir sprechen von Ereigniserkennung in Videos (engl. *Video Event Detection*) [ZHT06, MT08].

Weil verschiedene Methoden zur Positionsbestimmung sich auch verschieden verhalten, erzeugen sie verschiedene Trajektorien. Um die Qualität solcher Methoden zu messen, kann man außerdem diese Trajektorien miteinander und mit den manuell erstellten Trajektorien der tatsächlichen Position des Objektes vergleichen [NB03].

2.2 Handschrifterkennung

Ein weiteres Beispiel findet sich im Bereich der Handschrifterkennung. Wir können einen geschriebenen Buchstaben als Pfad des Stiftes durch den zweidimensionalen Raum, also als Trajektorie, betrachten. Um einen Buchstaben zu erkennen, muss dessen Trajektorie mit anderen Trajektorien, von denen wir wissen, welche Buchstaben sie darstellen, verglichen werden [VGD04].

Diese Anwendung stellt bereits andere Anforderungen an die Vergleichstechnik als das vorangehende Beispiel. In diesem Fall ist erwünscht, dass nur Form der Trajektorie entscheidend ist, also zum Beispiel zwei zueinander rotierte Trajektorien sich ähnlich sind. Diese Forderung der sogenannten Rotationsinvarianz (Definition 4.9) ist nur für bestimmte Anwendungsfälle sinnvoll. In Abschnitt 6.11 werden wir diesen Aspekt genauer beleuchten.

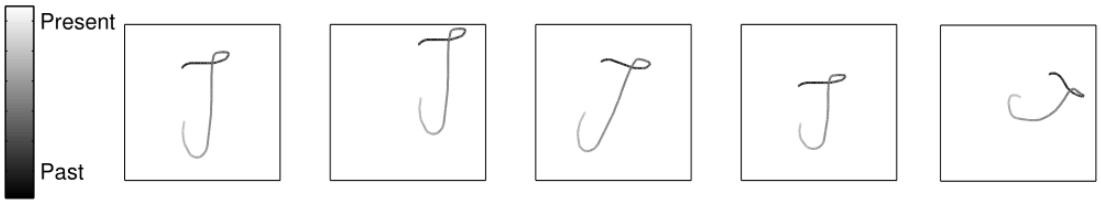


Abbildung 2.2: Trajektorien der Handschrift, die Translation, Rotation, Skalierung oder eine Kombination aus mehreren Transformationen erfahren haben [VGD04].

2.3 Forschung im Bereich Virtual Reality

Es gibt verschiedene Möglichkeiten, in einer virtuellen Welt zu navigieren. Man kann nicht nur einen Joystick benutzen, sondern auch die physischen Laufbewegungen einer realen Person auswerten (engl. *Walking-In-Place*) [SUS95]. Terziman et al. analysieren diese beiden Eingabemethoden und vergleichen dazu die Trajektorien, die jeweils bei einem Slalom entstehen [TMM+11].

2.4 Benutzeranalyse in Location-Based-Social-Networks

Zeichnet man die Standortdaten von Smartphonenutzern auf, erhält man Trajektorien ihrer Bewegungen. Diese Daten können vielfach verwendet werden, beispielsweise im Rahmen der Benutzeranalyse in Location-Based-Social-Networks (LBSN) [CLMP14, YLL+10]. Es ist zum Beispiel denkbar, dass eine Jogging-App die Bewegungsprofile ihrer Nutzer analysiert und Muster in Laufstrecken erkennt. Solche Daten können als Grundlage dafür dienen, die Ähnlichkeit der Nutzer zu bestimmen und ihnen so potentielle Joggingpartner vorzuschlagen. Auf eine dafür geeignete Technik gehen wir in Abschnitt 5.5.2 ein.

2.5 Datenverkehrsanalyse im Web

In allen bisherigen Beispielen lassen sich die Positionen als Koordinaten im euklidischen Raum, also einem üblichen Standardvektorraum über den reellen Zahlen, beschreiben. Dies muss nicht immer der Fall sein, wie in Abschnitt 6.1 behandelt werden wird. Dieses Beispiel verdeutlicht das und zeigt, dass die Berechnung der Ähnlichkeit von Trajektorien auch in weniger naheliegenden Bereichen nützlich sein kann.

Das Web zeichnet sich vor allem dadurch aus, dass Webseiten in Form von URLs auf andere verweisen. Diese Links bilden mit den Webseiten ein gigantisches Netz, das als Graph modelliert werden kann. Folgt man nun einigen der Links, hat man einen Pfad durch jenen Graphen zurückgelegt, den man als Trajektorie auffassen kann. Teil der Datenverkehrsanalyse im Web ist es, solche Aktivitäten zu untersuchen und zu ermitteln, welche Seiten der Nutzer besucht hat und wie lange er sich auf ihnen aufgehalten hat. Dabei kann es folglich zweckmäßig sein, die Ähnlichkeit von derartigen Trajektorien zu berechnen [TPN+09].

3 Problematik und Methode

In diesem Kapitel ordnen wir die vorliegende Arbeit im Hinblick auf existierende Literatur ein, leiten daraus die Problematik her und grenzen sie ein. Weiterhin erläutern wir die Vorgehensweise, die uns zu den Ergebnissen in den Kapiteln 5 und 6 führt.

3.1 Verwandte Arbeiten

Natürlich sind bei verwandten Arbeiten zuallererst alle Publikationen zu nennen, die Vergleichstechniken auf Trajektorien vorschlagen. Diese werden in gesammelter Form in Tabelle 7.1 zu finden sein. Als bekannte und verbreitete Techniken kann man dabei die euklidische Distanz, die Hausdorff- und die Fréchet-Distanz sowie Dynamic-Time-Warping (DTW) und Longest-Common-Subsequence (LCSS) ansehen. Diese Techniken stellen wir in Kapitel 5 im Detail zusammen mit ihrem Ursprung beziehungsweise ihrer zugrunde liegenden Publikation vor.

Es gibt allerdings auch Sekundärliteratur und generell solche, die sich auf abstrakterer Ebene mit dem Vergleich von Trajektorien beschäftigt. Meratnia und de By beschreiben in [MdB02] verschiedene Ansätze dafür, wie man große Mengen an Trajektorien aggregieren – also zusammenfassen – kann. Im Bezug auf Videoüberwachung (Abschnitt 2.1) vergleichen Zhang et al. in [ZHT06] Ähnlichkeitsmaße miteinander und kommen zu dem Ergebnis, dass einfachere euklidische Techniken aufgrund ihrer geringeren Komplexität geeigneter sind als DTW und LCSS. Auch Morris et al. führen in [MT09] einen solchen Vergleich für Anwendungen im Bereich der automatisierten Aktivitätserkennung durch, bescheinigen aber LCSS häufige Überlegenheit. In vielen Publikationen werden außerdem einzelne Schwächen von anderen Ähnlichkeitsmaßen ausgemacht und kritisiert, die das jeweils darin vorgestellte Ähnlichkeitsmaß nicht hat [CN04b, FGT07, YAS03, PF06, MP07].

Keine dieser Arbeiten reicht an den Umfang, die differenzierte Untersuchung der Ähnlichkeitsmaße und den Abstraktionsgrad der vorliegenden Arbeit heran. Auch Lehrbücher wie [ZZ11] verfügen in der Regel nur über eine beschränkte Sammlung an Ähnlichkeitsmaßen für Trajektorien und formulieren zudem keinen systematischen Vergleich zwischen

ihnen. Lediglich Wang und Ding et al. führen ausführliche Experimente durch, in denen sie zahlreiche Ähnlichkeitsmaße auf einer beachtlichen Menge von Datensätzen testen [WMD+13, DTS+08]. Ihr Resümee fällt sehr relativierend aus:

> „We found through experiments that there is no clear evidence that one similarity measure exists that is superior to others in the literature in terms of accuracy." [WMD+13, Abschn. 4.3]

In Anbetracht der breiten Anwendungsmöglichkeiten ist dies wenig verwunderlich. Welche Technik „besser" als eine andere ist, hängt davon ab, welche Anforderungen die konkrete Anwendung an das Ähnlichkeitsmaß stellt. Jene können – wie wir in Kapitel 2 gesehen haben – stark variieren.

3.2 Problematik

Durch die besagte Diversität der Anwendungen haben sich stark unterschiedliche Vergleichstechniken entwickelt. Einige davon erfüllen bestimmte Anforderungen, die andere nicht erfüllen, und umgekehrt. Darüber hinaus kann jede Technik in verschiedenen Bereichen eingesetzt werden, die zum Zeitpunkt ihrer Definition nicht absehbar sind. Der Abstraktionsgrad dieser Arbeit lässt deswegen keine absoluten Wertungen über die vorgestellten Techniken zu.

Was nötig wäre, um Vergleichstechniken anwendungsunabhängig und objektiv gegeneinander abwägen und sortieren zu können, sind neutrale Maßstäbe, anhand derer sie sich ihrerseits vergleichen lassen. Solche Maßstäbe würden es ermöglichen, eine Systematik einzuführen, um Ähnlichkeitsmaße auf Trajektorien schon vor ihrer Anwendung zu bewerten und zu ordnen. Die Auswahl eines für eine bestimmte Anwendung geeigneten Ähnlichkeitsmaßes aus einer unüberschaubaren Vielfalt würde enorm vereinfacht und beschleunigt werden, weil man anhand der Anforderungen der Anwendung in der Systematik navigieren könnte und schnell ungeeignete Techniken ausschließen sowie geeignete eingrenzen könnte.

Auch nach eingehender Recherche sind uns keine solchen Maßstäbe bekannt. Aufgrund der Vielzahl der Anwendungsmöglichkeiten und der erläuterten Relevanz – inbesondere durch die wachsende Menge spatiotemporaler Daten – erscheint es dringend nötig, solche Maßstäbe zu definieren und somit eine Ordnung für existierende und noch nicht existierende Methoden herzustellen. Dies umzusetzen, ist das Ziel dieser Arbeit.

3.3 Methode

Um die fehlenden Maßstäbe für den Vergleich von Vergleichstechniken sinnvoll definieren zu können, muss man deren Gemeinsamkeiten und Unterschiede ausmachen und die sich daraus ergebenden Eigenschaften beziehungsweise Vor- oder Nachteile nachvollziehen. Damit die Maßstäbe generisch sind und sich auf möglichst jedes Ähnlichkeitsmaß anwenden lassen, ist es außerdem wichtig, eine gewisse Breite an unterschiedlichen Methoden zu betrachten.

Folglich ist es unerlässlich, sich zunächst ausgiebig mit existierender Literatur auseinanderzusetzen. Dies geschah anfangs mit Lehrbüchern [ZZ11] und dann durch Suche nach Publikationen mit thematisch eingrenzenden Stichwörtern wie „trajectory“, „comparison“, „similarity“, „distance“, „moving object“ oder „time warping“. Durch Querverweise in diesen Publikationen ergibt sich rasch eine große Menge an Material. Bei dessen Sichtung finden sich Literatur mit vergleichbarer Motivation [MT09], hilfreiche Dissertationen, die Problematiken erläutern [Che05] sowie neue unbekanntere Ansätze, die bekannte verbessern [MP07] oder ganz neue Konzepte vorstellen und so zur Abstraktion des bestehenden Verständnisses zwingen [TPN+09]. Es findet sich auch weniger zielgerichtete Literatur, die sich nur im erweiterten Sinne mit dem Trajektorienvergleich befasst [LHW07], die zwar thematisch treffende Problematiken enthält, aber so außergewöhnliche Ansätze verfolgt, dass deren genauere Betrachtung in andere Fachgebiete und damit zu weit führt [Por04], oder die zwar interessante Lösungen versprechen, aber bei denen sich nach genauerer Analyse zeigt, dass sie entweder leicht andere Problematiken behandeln [BYÖ97] oder sich lediglich auf bereits bekannte andere Ansätze berufen [DN05].

Kapitel 5 spiegelt die Substanz dieser Recherche wider, weil dort für relevant befundene Techniken in aufgearbeiteter Form vorgestellt werden. Sie sind sorgfältig aufeinander abgestimmt und explizit als Ähnlichkeitsmaß auf Trajektorien definiert. Man gewinnt einen umfassenden Eindruck von existierenden Ansätzen im Zusammenhang mit dem Vergleich von Trajektorien. Damit dies gelingt, sind eine einheitliche Notation, wohldefinierte Begriffe sowie die Vermittlung eines gewissen Verständnisses von Zusammenhängen zwischen Techniken hilfreich und wichtig. Daher befasst sich das nächste Kapitel 4 mit diesen Grundlagen und bereitet so auf Kapitel 5 vor.

Die zuvor in Abschnitt 3.2 geschilderte Problematik wird im Anschluss in Kapitel 6 behandelt. Die erwähnten fehlenden Maßstäbe für den Vergleich von Vergleichstechniken führen wir in Form von Klassifikationen ein. Jede dieser Klassifikationen erlaubt die eindeutige Zuordnung eines Ähnlichkeitsmaßes zu einer von mehreren Klassen. Wir leiten die Klassen von den beobachtbaren Eigenschaften der zuvor betrachteten Ähnlichkeitsmaße

ab, sodass jene sich tatsächlich aus den inhärenten Charakteristika dieser ergeben. Das so definierte Klassensystem gewährleistet einen systematischen, anwendungsunabhängigen Vergleich von Vergleichstechniken für Trajektorien.

Wir definieren in Kapitel 6 indes nicht nur die Klassengrenzen, indem wir existierende Unterschiede zwischen Ähnlichkeitsmaßen in Kategorien benennen, sondern machen von jeder Klassifikation unmittelbar Gebrauch, indem wir in Kapitel 5 gesammelte Ähnlichkeitsmaße einordnen und diese Einordnung erklären. Hierfür sind speziell die expliziten und einheitlichen Definitionen der Ähnlichkeitsmaße hilfreich. Auf diese Weise demonstrieren wir konkret die Praxistauglichkeit und Sinnhaftigkeit der jeweiligen Klassifikation. Das Resultat ist ein Klassensystem, nach dem sich Ähnlichkeitsmaße und damit Vergleichstechniken für Trajektorien objektiv ordnen lassen.

4 Grundlagen

Dieses Kapitel behandelt technische Grundlagen der Thematik und bereitet so auf die darauf folgenden Kapitel vor. Aus Gründen der Konsistenz und des Verständnisses ist dies unabdingbar.

4.1 Terminologie

Um unterschiedliche Methoden verstehen, einordnen und miteinander vergleichen zu können, sowie Missverständnissen vorzubeugen, müssen verwendete Begriffe und Termini präzise und explizit definiert werden. Das ist umso wichtiger, wenn verschiedene Autoren ein und denselben Terminus mit unterschiedlicher Bedeutung verwenden. Dieser Abschnitt widmet sich deswegen der Begriffsbildung. Im weiteren Verlauf der Arbeit werden wir uns auf die Definitionen aus diesem Abschnitt beziehen. Sie orientieren sich größtenteils an existierenden Definitionen aus der Literatur, um Konventionen nach Möglichkeit nicht zu brechen.

4.1.1 Trajektorien und Zeitreihen

Die Begriffe der *Trajektorie* und der *Zeitreihe* hängen eng miteinander zusammen. Während Zeitreihen in der Wissenschaft jedoch schon lange weit verbreitet und gut erforscht sind, sind Trajektorien in der Informatik ein relativ junges Forschungsgebiet. Für beide gibt es unterschiedliche Definitionen und auch über ihren Zusammenhang herrscht keine Einigkeit. Manchmal werden sie strikt voneinander getrennt, manchmal werden Trajektorien als eine Unterart von Zeitreihen definiert [ZHT06, Kap. 2], manchmal werden sie sogar synonym verwendet [Che05]. Daher bezieht sich die verwendete Literatur nicht nur auf Trajektorien, sondern in vielen Fällen auch auf Zeitreihen.

Eine Zeitreihe wird üblicherweise erzeugt, indem wiederholt zu bestimmten Zeitpunkten Daten von einem System unserer Betrachtung erhoben werden (engl. *sampling*). Daraus resultiert die folgende Definition:

Definition 4.1 (Zeitreihe) *Eine **Zeitreihe** T (engl.* time series*) ist eine Folge (auch* Sequenz*) von Daten, wobei ein Datum $\mathtt{d}_i$ jeweils mit einem Zeitstempel $\mathtt{t}_i$ versehen ist:*

$$\mathsf{T} = \big(\langle \mathtt{d}_1, \mathtt{t}_1\rangle, \langle \mathtt{d}_2, \mathtt{t}_2\rangle, \dots \langle \mathtt{d}_n, \mathtt{t}_n\rangle\big)$$

und die Zeitstempel streng monoton steigend sind: $\forall\, i \in \{1, \dots n-1\}\colon \mathtt{t}_i < \mathtt{t}_{i+1}$.

Wir können annehmen, dass die Zeit als nicht-negative reelle Zahl ausgedrückt wird: $\mathtt{t}_i \in \mathbb{R}^+$ wobei $\mathbb{R}^+ = \{x \in \mathbb{R} \mid x \geqslant 0\}$. Außerdem werden die Daten oft in regelmäßigen Abständen erhoben, sodass ihre Zeitstempel **äquidistant** sind, also ihre Distanz nicht variiert: $\forall\, i \in \{1, \dots n-1\}\colon \mathtt{t}_{i+1} - \mathtt{t}_i = c$ für eine Konstante $c \in \mathbb{R}^+$. Die Distanz zwischen solchen Zeitstempeln heißt **Samplingrate**.

Weil ein Datum von einer reellen Zahl, die eine Spannung in Volt repräsentiert, über einen Aktienindex bis zu politischen Umfragewerten alles sein kann, ist die Definition der Zeitreihe sehr generisch. Im Gegensatz dazu implizieren wir mit dem Begriff der *Trajektorie*, dass die Daten Punkte in einem Raum (*Positionen*) sind, allen voran der euklidische Raum. Tatsächlich soll eine Trajektorie den Pfad eines bewegten Objektes durch den Raum darstellen.

Definition 4.2 (Kontinuierliche Trajektorie) *Eine **kontinuierliche Trajektorie** τ ist eine Funktion $\tau : \mathbb{R}^+ \to \mathsf{P}$, die die Zeit $t \in \mathbb{R}^+$ auf Positionen $p \in \mathsf{P}$ abbildet.*

Wir definieren an dieser Stelle ausdrücklich nicht, was genau eine *Position* ist, also welche Form die Menge aller Positionen P hat. Nichtsdestotrotz haben wir es in vielen Fällen mit Trajektorien im zwei- oder dreidimensionalen euklidischen Raum zu tun, sodass eine Position aus den üblichen Koordinaten $p = \langle x, y\rangle$ beziehungsweise $p = \langle x, y, z\rangle$ besteht, wobei $x, y, z \in \mathbb{R}$. Wir sprechen dann auch von **Punkten** statt Positionen. Allerdings gibt es andere Wege, Positionen zu definieren, zum Beispiel als positionellen Identifikator in Kombination mit einem Versatz (engl. *offset*) [HKL06, Abschn. 2.2]. Einige Autoren sehen eine Position abstrakt als Element einer definierten Menge von geographischen Positionen an und konkretisieren diese gar nicht weiter [CLMP14]. Im Allgemeinen kann eine Position als Vektor einer festen Arität $\geqslant 2$ verstanden werden. Der Ausschluss von eindimensionalen Positionen stellt eine Abgrenzung zu Zeitreihen dar.

In der Realität werden keine kontinuierlichen Daten erfasst, weil es nur eine endliche Menge an Beobachtungen des bewegten Objektes gibt, also eine endliche Teilmenge der reellen kontinuierlichen Trajektorie. Diese Menge an Beobachtungen konstituiert die diskreten Daten der Trajektorie zu endlich vielen Zeitpunkten. Tatsächlich gehen viele Autoren von Anfang an davon aus, dass Zeit diskret ist [WMD+13, Kap. 2].

Definition 4.3 (Trajektorie) *Eine (diskrete)* ***Trajektorie*** τ *ist eine Sequenz von Positionen, wobei eine Position* p_i *jeweils mit einem Zeitstempel* t_i *versehen ist:*

$$\tau = \big(\langle p_1, t_1\rangle, \langle p_2, t_2\rangle, \dots \langle p_n, t_n\rangle\big)$$

und die Zeitstempel streng monoton steigend sind: $\forall\, i \in \{1, \dots n-1\}\colon t_i < t_{i+1}$.

Wenn man die Semantik der diskreten Punkte einer Trajektorie bedenkt, also wie sie zustande kommen, ist es vernünftig anzunehmen, dass sie dazu verwendet werden können, die reelle kontinuierliche Trajektorie zu rekonstruieren. Dies wird mit Interpolation erreicht. Wie genau dies geschieht, werden wir in Abschnitt 4.2.2 genauer beleuchten.

Offenkundig weisen die Definitionen der diskreten Trajektorie und die der Zeitreihe starke Parallelen auf. Gängige Unterscheidungen betreffen:

- die Eindimensionalität der Daten bei der Zeitreihe und Multidimensionalität ebenjener bei der Trajektorie [MdB02, Che05],
- die Annahme der Kontinuität der ursprünglichen Daten bei der diskreten Trajektorie [WMD+13] und
- die Semantik der Pfade von bewegten Objekten der Trajektorie [Por04].

Im engeren Sinne ist eine Trajektorie als Pfad eines bewegten Objektes immer ein kontinuierliches Objekt und jede diskrete Trajektorie nur eine Darstellung durch ein **Sampling** von ihr. Den Terminus *Zeitreihe* verwenden wir in dieser Arbeit nur, wenn eine bestimmte Technik dies aufgrund ihres Hintergrundes nahelegt, und sprechen ansonsten von *Trajektorien.*

Wir bezeichnen ein einzelnes Tupel aus Position und Zeitstempel $\tau[i] = \langle p_i, t_i\rangle$ als **Glied** der Trajektorie mit **Index** i. Manchmal ist mit dieser Notation, sofern der Kontext dies nahelegt, auch nur die Position gemeint: $\tau[i] = p_i$. In Ergänzung dazu bezeichnen wir mit $t_{\sigma,i}$ den Zeitstempel des Gliedes mit Index i einer Trajektorie σ. Genau wie bei kontinuierlichen Trajektorien, möchten wir auf die Position zu einem bestimmten Zeitpunkt zugreifen können. Wir betrachten τ daher ebenfalls als partielle Funktion und definieren $\tau(t_i) = p_i$ für jene Zeitstempel t_i, die in τ vorkommen. In dieser Notation ist das Argument also der Wert des Zeitstempels, nicht sein Index.

Als **Resampling** einer im Zeitintervall von t_1 bis t_n definierten diskreten Trajektorie τ mit n Gliedern bezeichnen wir jede andere Trajektorie τ' mit m Gliedern, für die gilt, dass die ersten und letzten Glieder jeweils übereinstimmen ($\tau[1] = \tau'[1] \wedge \tau[n] = \tau'[m]$) und die repräsentierte kontinuierliche Trajektorie dieselbe ist: $\forall\, t \in [t_1 \dots t_n]\colon \tau(t) = \tau'(t)$.

Das erste Glied $\tau[1]$ einer Trajektorie τ oder dessen Position bezeichnen wir mit $\text{head}(\tau)$. Die Trajektorie ohne das erste Glied $\big(\tau[2], \tau[3], \ldots \tau[n]\big)$ bezeichnen wir mit $\text{tail}(\tau)$.

Insbesondere im Falle von Punkten im euklidischen Raum ist es außerdem üblich, Trajektorien als Sequenz von Tupeln der Form $\langle x_i, y_i, t_i \rangle$ beziehungsweise $\langle x_i, y_i, z_i, t_i \rangle$ zu definieren, anstatt die Tupel unnötigerweise zu schachteln [Wol02]. Wenn die Zeitstempel äquidistant sind, kann man sie sogar einsparen und Trajektorien schlicht als Sequenz von Positionen definieren. Die Zeitinformation ist dann implizit [VKG02, PF06]. Im Gegensatz dazu definieren Needham und Boyle Trajektorien explizit als Sequenz von Positionen und Zeitstempeln und unterscheiden sie so von einem **Pfad** (engl. *path*), den sie als Sequenz von Positionen ohne Zeitstempel definieren [DTS+08, Abschn. 2.1].

Definition 4.4 (Segment, Subtrajektorie, Strecke) *Sei* $\big(\tau[1], \ldots \tau[n]\big)$ *eine Trajektorie. Eine Teilfolge* $\big(\tau[i], \ldots \tau[j]\big)$ *einer Trajektorie von Index* i *bis* j*, wobei* $1 \leqslant i < j \leqslant n$*, nennen wir* ***Segment*** *der Trajektorie oder* ***Subtrajektorie****. Ein Segment minimaler Länge zwischen zwei adjazenten Gliedern, also* $j = i + 1$*, heißt* ***Strecke*** *(engl.* line segment*) der Trajektorie.*

Diese nicht so stark verbreitete Terminologie der Segmente wird zum Beispiel in dieser Form in [WMD+13] verwendet. Bei Betrachtung der Literatur fällt auf, dass die Art und Weise, die Länge oder Größe von Trajektorien zu quantifizieren, keinesfalls eindeutig ist. Deswegen legen wir folgende Begriffe fest:

Definition 4.5 (Länge, Akkumulat, Verschiebung) *Sei* d *eine Distanzfunktion*[1] *auf Punkten. Die* ***Länge*** *(engl.* length*) einer (diskreten) Trajektorie oder Zeitreihe ist die Anzahl der Glieder ihrer Sequenz (Die Anzahl der Segmente plus eins). Das* ***Akkumulat*** *einer Trajektorie ist die Summe der Längen ihrer Strecken:* $\sum_{i=1}^{n-1} d(\tau[i], \tau[i+1])$*.*[2] *Die* ***Verschiebung*** *(engl.* displacement*) einer (diskreten) Trajektorie ist die Distanz von ihrem ersten Glied zu ihrem letzten Glied:* $d(\tau[1], \tau[n])$*.*

Man beachte, dass wir Länge und Akkumulat auf diese Weise nicht für kontinuierliche Trajektorien definieren können. Trotzdem ist klar, dass beide als die Länge des Weges des tatsächlichen Pfades von einem Ende bis zum anderen definiert werden würden. Die Definition der Verschiebung wäre analog zur diskreten Trajektorie. Man beachte darüber hinaus, dass mit Distanz zweier Punkte nicht notwendigerweise räumliche Distanz gemeint sein muss, sondern auch zeitliche oder spatiotemporale Distanz gemeint sein kann.

[1] Wir werden in Definition 4.7 genauer spezifizieren, was das heißt.

[2] Porikli definiert unser Akkumulat als *length* und unsere Länge als *duration* [Por04, Kap. 1 und 3]. Wir übernehmen dies jedoch nicht, weil unsere Definition der Länge deutlich verbreiteter ist.

4.1.2 Ähnlichkeit und Distanz

In diesem Abschnitt definieren wir zwei zentrale Begriffe: *Ähnlichkeitsmaß* und *Distanz*. Da in dem Wort „Ähnlichkeitsmaß“ das Wort „Maß“ steckt, sei an dieser Stelle darauf hingewiesen, dass ein Ähnlichkeitsmaß nicht unbedingt ein Maß sein muss. Nach der üblichen Definition ist ein **Maß** (engl. *measure*) eine Abbildung von einer Sigma-Algebra Σ über einer Menge X in die erweiterten reellen Zahlen $\overline{\mathbb{R}}$, die Eigenschaften wie Nicht-Negativität und Sigma-Additivität erfüllt. Die **erweiterten reellen Zahlen** bezeichnen die Menge $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$, die auch unendliche Elemente enthält. Interessanterweise kann ein solches Maß nicht auf beliebigen Teilmengen von $\mathcal{P}(\mathbb{R})$ definiert werden, wie Vitali 1905 gezeigt hat [Vit08]. Um dieses sogenannte Maßproblem zu umgehen, werden zum Beispiel nur Maße auf bestimmten Mengen definiert, wie Lebesgue es erstmals tat [Leb02]. Weitere Gedanken dazu führen uns in die Maßtheorie und sprengen den Rahmen dieser Arbeit. Hiermit soll aber angedeutet sein, dass es wichtig ist, solch grundlegende Begriffe zu hinterfragen. Wir nehmen Abstand von der strikten Definition des Maßes und definieren für unsere Zwecke das *Ähnlichkeitsmaß*, das auch gemeint sein soll, wenn wir von *Ähnlichkeit* sprechen. Es soll im Gegensatz zum Maß zum Beispiel auch negative Werte erlauben.

Definition 4.6 (Ähnlichkeitsmaß) *Ein **Ähnlichkeitsmaß** (engl.* similarity measure *oder* similarity function*) ist eine Funktion $s : M \times M \to \overline{\mathbb{R}}$, die die Ähnlichkeit zweier Elemente der gleichen Menge M als erweiterte reelle Zahl quantifiziert.*

Im Allgemeinen erwarten wir, dass ein Ähnlichkeitsmaß umso größere Werte für zwei Objekte annimmt, je ähnlicher sich die beiden sind. Das Ähnlichkeitsmaß für zwei ähnliche Objekte ist also eine größere Zahl als das für zwei unähnliche Objekte. In dieser Hinsicht kann *Ähnlichkeitsmaß* als das Inverse von *Distanz*, *Abstand* oder *Unähnlichkeitsmaß* verstanden werden, für die sich dies genau umgekehrt verhält. Wir bezeichnen das generelle Inverse eines Ähnlichkeitsmaßes in diesem Sinne als **Unähnlichkeitsmaß**, während wir uns vorbehalten, mit dem allgemeineren Begriff *Ähnlichkeitsmaß* auch Unähnlichkeitsmaße einzuschließen. An den Begriff der **Distanz** stellen wir strengere Anforderungen:

Definition 4.7 (Distanz, Metrik) *Eine **Distanz**, **Distanzfunktion** oder **Metrik** auf einer Menge M ist eine Funktion $d : M \times M \to \mathbb{R}^+ \cap \overline{\mathbb{R}} = [0, \infty]$, die für alle $x, y, z \in M$ die folgenden Bedingungen erfüllt:*

- ***Nicht-Negativität:*** $d(x, y) \geqslant 0$
- ***Identitätsprinzip:*** $d(x, y) = 0 \Leftrightarrow x = y$
- ***Symmetrie:*** $d(x, y) = d(y, x)$
- ***Dreiecksungleichung:*** $d(x, y) \leqslant d(x, z) + d(z, y)$

Die obige Definition wäre äquivalent, wenn wir die erste Bedingung auslassen würden, da sie aus der zweiten und vierten Bedingung folgt. Wenn nur die ersten beiden Bedingungen erfüllt sind, heißt die Funktion **positiv definit**. Wenn wir die dritte Bedingung der Symmetrie auslassen, heißt die Funktion **Quasimetrik**.[3] Wenn wir die vierte Bedingung der Dreiecksungleichung auslassen, heißt die Funktion **Semimetrik**. Eine Menge M mit einer Metrik d heißt **metrischer Raum** (M, d).

Im Folgenden benennen wir Unähnlichkeitsmaße konventionell mit d (für engl. *distance*), auch wenn diese nicht immer die Eigenschaften einer Metrik erfüllen. Ähnlichkeitsmaße im engeren Sinne, die also mit steigender Ähnlichkeit wachsen, benennen wir mit s.

4.1.3 Invarianz unter Transformationen

Der Begriff der Invarianz unter geometrischen Transformationen ist einer expliziten Definition würdig.

Eine Menge M eines Raumes erlaubt potentiell die Parallelverschiebung oder **Translation** ihrer Elemente. Damit ist eine geometrische Transformation $\oplus : M \times M' \to M$ gemeint, die ein Element $m \in M$ um einen Verschiebungsvektor $m' \in M'$ gleichmäßig so verschiebt, dass Längen und Winkel erhalten bleiben. Zum Beispiel können Trajektorien im zweidimensionalen euklidischen Raum um Vektoren aus $\mathbb{R}^2$ verschoben werden. Neben der Translation sind weitere Transformationen möglich, unter anderem die **Rotation**, bei der Elemente von M um einen Winkel α rotiert werden: $\mathrm{rot} : M \times [-\pi, \pi] \to M$, und die **Skalierung**, bei der Elemente um einen festen Skalar $\lambda \in M'$ vergrößert oder verkleinert werden: $* : M \times M' \to M$. Diese Transformationen erlauben uns folgende Definitionen:

Definition 4.8 (Translationsinvarianz) *Ein Ähnlichkeitsmaß d heißt **translationsinvariant** genau dann, wenn $d(x, y) = d(x \oplus a, y)$ für alle $x, y \in M, a \in M'$ gilt.*

Wir werden eine Reihe weiterer Begriffe im Zusammenhang mit dem Umgang mit Trajektorien benutzen, die einer expliziten Definition würdig sind. Das umfasst zum Beispiel Transformationen:

Definition 4.9 (Rotationsinvarianz) *Ein Ähnlichkeitsmaß d heißt **rotationsinvariant** genau dann, wenn $d(x, y) = d(x \,\mathrm{rot}\, \alpha, y)$ für alle $x, y \in M, \alpha \in [-\pi, \pi]$ gilt.*

Definition 4.10 (Skalierungsinvarianz) *Ein Ähnlichkeitsmaß d heißt **skalierungsinvariant** genau dann, wenn $d(x, y) = d(x * \lambda, y)$ für alle $x, y \in M, \lambda \in M'$ gilt.*

[3] Für den Begriff des Ähnlichkeitsmaßes fordern wir keine Symmetrie.

Wenn man von Translation, Rotation und Skalierung absieht, bleibt schließlich die sogenannte Form einer Trajektorie übrig. Um genau zu sein, haben zwei Trajektorien die gleiche **Form**, wenn sie unter Verwendung eines translations-, rotations- und skalierungsinvarianten Unähnlichkeitsmaßes identisch sind.[4]

Es existieren auch alternative Definitionen dieser Begriffe, bei denen jeweils beide Parameter x und y verschoben, rotiert beziehungsweise skaliert werden. Im Falle der Translationsinvarianz bedeutete dies sinngemäß, dass es keinen Unterschied macht, *wo* im Raum die Berechnung stattfindet. Im Gegensatz dazu geht es bei obiger Definition um eine Veränderung der Ähnlichkeit *zwischen tatsächlich verschobenen Elementen*. Die alternative Definition wird allerdings von der euklidischen Distanz und fast allen darauf aufbauenden Ähnlichkeitsmaßen erfüllt und ist für uns nicht von primärem Interesse. Auch die Rotationsinvarianz im alternativen Sinne wird von fast jedem Ähnlichkeitsmaß erfüllt. Nur skalierungsinvariant bei der Skalierung beider Trajektorien sind viele Ähnlichkeitsmaße nicht.

4.2 Weiterführende Techniken

Rechnet man mit Trajektorien, hat man es mit einer Reihe von Methoden zu tun, die zwar nicht den Vergleich zwischen ihnen direkt betreffen, aber so stark mit ihm zusammenhängen, dass wir sie wegen ihrer Relevanz nicht unerwähnt lassen können.

4.2.1 Normalisierung

Üblicherweise durchlaufen die Daten der Trajektorien oder Zeitreihen eine **Vorverarbeitung** (engl. *preprocessing*). Dazu kann die Normalisierung im Sinne der Statistik gehören. Daten, mit denen dies geschehen ist, bezeichnen wir als **normalisiert**. Man kann zum Beispiel wie folgt verfahren: Für jede Dimension der Daten wird das arithmetische Mittel μ und die Standardabweichung σ über alle Glieder gebildet. Anschließend werden die Werte d_i jeweils mit $(d_i - \mu)/\sigma$ ersetzt [GK95].

Mit Normalisierung kann aber jegliche Form von Vorverarbeitung gemeint sein, die die Daten in irgendeiner Form vereinheitlicht oder in einen konsistenten, zweckmäßigen Zustand überführt. Zum Beispiel werden wir in Abschnitt 5.3.12 eine *zeitlich normalisierte Trajektorie* definieren, die durch eine Normalisierung äquidistante Zeitstempel haben wird.

[4] Prägnant ausgedrückt, wenn ihre transformationsinvariante Distanz (Definition 5.26) gleich Null ist.

4.2.2 Interpolation

In Abschnitt 4.1.1 haben wir erläutert, dass aus diskreten Trajektorien durch **Interpolation** kontinuierliche gewonnen werden können. Dafür existieren mehrere Methoden. Die mit Abstand gängigste davon ist die lineare Interpolation [AKK+06, Kap. 3]:

Definition 4.11 (Piecewise-Linear-Approximation) *Sei* $\tau = \big(\langle p_1, t_1\rangle, \langle p_2, t_2\rangle, \dots \langle p_n, t_n\rangle\big)$ *eine Trajektorie. Wir können die Position zum Zeitpunkt* t *mit der* ***Piecewise-Linear-Approximation*** *(PLA) wie folgt bestimmen:*

$$\tau(t) = \begin{cases} \tau(t) & , t \textit{ kommt in } \tau \textit{ vor} \\ \frac{t_{i+1}-t}{t_{i+1}-t_i}\tau(t_i) + \frac{t-t_i}{t_{i+1}-t_i}\tau(t_{i+1}) & , \textit{sonst} \end{cases}$$

wobei i *so gewählt sei, dass* $t_i < t < t_{i+1}$.

Es wird also eine geradlinige Strecke zwischen zwei Gliedern angenommen, die als Annäherung an die tatsächliche Position der kontinuierlichen Trajektorie fungiert. Andere Möglichkeiten der Interpolation umfassen die Annäherung durch einen konstanten Wert oder – etwas komplexer – durch ein Polynom.

Die Interpolation einer Position kann nützlich sein, um von einer Trajektorie nicht alle ursprünglich erhobenen Daten speichern zu müssen. Das führt uns zum nächsten Abschnitt.

4.2.3 Datenreduktion

Da Trajektorien oft in großer Zahl erzeugt werden, versucht man in der Regel, die Datenmenge zu reduzieren, um einerseits Speicherplatz zu sparen und andererseits Berechnungen darauf effizient oder gar erst durchführbar zu machen. Wenn die Samplingrate hoch ist, sind die Trajektorien lang und die Menge an Daten ist groß. Für den vorliegenden Anwendungszweck ist jene unter Umständen deutlich feingranularer als nötig. In diesem Fall können wir die Trajektorien durch andere vereinfachte Trajektorien geringerer Länge ersetzen. Man geht dann davon aus, dass man die ursprünglichen Daten aus den vereinfachten Daten hinreichend interpolieren kann. Dafür prädestiniert ist der Douglas-Peucker-Algorithmus [HG97, GTW+10]. Er entfernt schlicht Glieder aus der Trajektorie, solange die dadurch entstehende Vereinfachung eine gegebene Schranke für den Fehler nicht überschreitet.

Um die Daten von Zeitreihen und Trajektorien zu reduzieren, gibt es eine Vielzahl von Techniken [DTS+08, Kap. 1]:

- Single-Value-Decomposition (SVD) [FRM94]
- Diskrete Fourier-Transformation (DFT) [AFS93]
- Diskrete Wavelet-Transformation (DWT) [CF99]
- Piecewise-Aggregate-Approximation (PAA) [KCPM01]
- Piecewise-Constant-Approximation (PCA)
- Adaptive-Piecewise-Constant-Approximation (APCA) [Keo06]
- Chebyshev-Polynomials (CHEB) [CN04a]
- Symbolic-Aggregate-Approximation (SAX) [LKWL07]
- Indexable-Piecewise-Linear-Approximation (IPLA) [CCL+07]

Piecewise-Aggregate-Approximation zum Beispiel erzeugt Trajektorien beziehungsweise Zeitreihen, die näherungsweise mit den Originaldaten übereinstimmen und die Eigenschaft haben, eine untere Schranke (engl. *lower bound*) für sie zu sein [KCPM01]. PAA ist ursprünglich auf eindimensionale Daten beschränkt, kann aber auf mehrdimensionale Daten erweitert werden. Dies findet Anwendung in der Erstellung von Indexstrukturen zur effizienten Suche nach Trajektorien in einer Datenbank [YAS03].

Die aufgelisteten Techniken sind sich im Ergebnis alle sehr ähnlich [WMD+13, Kap. 3]. Dies führt dazu, dass sie sich alle für den Aufbau von Indexstrukturen eignen, solange sie die erwähnte Eigenschaft der unteren Schranke haben:

> „In [FRM94], the authors propose the GEMINI framework, that allows to incorporate any dimensionality reduction method into efficient indexing, as long as the distance function on the reduced feature space fulfills the lower bounding property.“ [AKK+06, Abschn. 2]

4.2.4 Merkmalsräume

Eine oft vielversprechende Methode ist es, die Ähnlichkeit von Trajektorien nicht direkt im Raum, in dem sie liegen, zu berechnen, sondern sie in einen anderen Raum zu überführen, der gewünschte Vorteile – etwa die Rotationsinvarianz für Handschrifterkennung (Abschnitt 2.2) – bietet. Weil die Repräsentationen der Trajektorien in solchen Räumen bestimmte Merkmale der ursprünglichen Trajektorien tragen, nennt man diese **Merkmalsräume** (engl. *feature spaces*).

Einige der im nächsten Kapitel vorgestellten Ansätze arbeiten mit Merkmalsräumen, zum Beispiel AAL-Warping (Abschnitt 5.3.16) oder die Spatial-Assembling-Distance (Abschnitt 5.4.1). Da sich aus ihnen aber völlig neue Möglichkeiten und Schwierigkeiten ergeben und sie sich schlechter vergleichen lassen, richten wir unser Hauptaugenmerk auf Techniken, die ohne Merkmalsräume auskommen.

5 Systematische Darstellung von Vergleichstechniken

Wie in den ersten Kapiteln erwähnt, gibt es zahlreiche stark variierende Möglichkeiten, Trajektorien zu vergleichen. In diesem Kapitel stellen wir die bekanntesten und einige eher unbekannte, aber interessante Ansätze vor. Zu Ansätzen in diesem Sinne zählen nicht nur Ähnlichkeitsmaße auf Trajektorien, sondern auch dafür benötigte grundlegendere Techniken, sowie weiterführende Ideen, die über den schlichten Entwurf von Ähnlichkeitsmaßen hinaus gehen. Die Liste der in diesem Kapitel angeführten Techniken erhebt keinen Anspruch auf Vollständigkeit. Vielmehr soll sie in die Details des Trajektorienvergleiches einführen und einen Überblick über existierende Methoden verschaffen. Die Liste hebt sich aus folgenden Gründen von einer einfachen Sammlung ab: Erstens sind die Techniken in einer didaktischen sinnvollen Reihenfolge vorgestellt, sodass sie stringent aufeinander aufbauen. Zweitens sind die zugrunde liegenden Publikationen so kurz wie möglich, aber so ausführlich wie für das Verständnis nötig zusammengefasst. Drittens sind die Begriffe angepasst und die Notationen der Definitionen von den Publikationen abstrahiert und vereinheitlicht. Nicht zuletzt ist die Liste recht umfangreich, was die Bandbreite des Trajektorienvergleiches unterstreicht und so in der Literatur bisher nicht zu finden ist.

Allgemein kann man geometrische Ähnlichkeitsmaße in drei Kategorien einordnen: Erstens solche, die Positionen mit Positionen vergleichen, zweitens solche, die Positionen mit Trajektorien vergleichen, und drittens solche, die Trajektorien mit Trajektorien vergleichen. Letzteren schenken wir naheliegenderweise den Großteil unserer Aufmerksamkeit. Die ersten beiden Abschnitte dieses Kapitels beschäftigen sich mit den ersten beiden dieser Kategorien. Anschließend stellen wir Techniken für den Vergleich von Trajektorien vor. Ansätzen, die aus diversen Gründen aus dem Rahmen fallen, widmen wir einen eigenen Abschnitt. Ein zusätzlicher Abschnitt behandelt Objekte, die keine Trajektorien sind, aber deren Vergleich thematisch stark mit dem von Trajektorien zusammenhängt.

Unabhängig von Trajektorien gibt es eine Fülle von Ähnlichkeitsmaßen und Metriken für multidimensionale Daten oder Daten, die in irgendeiner Form nicht trivial sind. *Trivial*

meint in diesem Zusammenhang, dass deren Distanz auf der Hand liegt, weil sie sich zum Beispiel wie die reellen Zahlen ohne zusätzliche Definition in eine Kardinalskala fügen. In manchen Fällen lassen sich diese Ähnlichkeitsmaße unverändert für Trajektorien verwenden. In einigen anderen Fällen stammen die grundlegenden Ideen für den Vergleich aus einem anderen Fachgebiet und können in veränderter Form für Trajektorien verwendet werden. Zum Entwurf von Ähnlichkeitsmaßen auf Zeitreihen und Trajektorien schreiben Ding et al.:

> „[...] unlike canonical data types, e.g., nominal or ordinal variables, where the distance definition is straightforward, the distance between time series needs to be carefully defined in order to reflect the underlying (dis)similarity of such data." [DTS+08, Kap. 1]

5.1 Positionsvergleich

Der Vergleich von einzelnen Positionen ist aus zwei Gründen für den Vergleich von Trajektorien elementar. Zum einen ist seine Definition nicht immer offenkundig. Sind die Positionen etwa keine kartesischen Koordinaten, sondern Knoten eines Graphen, die aber wiederum mit kartesischen Koordinaten versehen sind, gibt es mehrere Möglichkeiten, einen Abstand zwischen ihnen zu definieren: als Kostenfunktion im Graphen, euklidisch zwischen den Koordinaten oder als Kombination aus beidem. Zum anderen bildet er fast immer die Basis für den Vergleich mit Trajektorien, da diese aus Positionen bestehen und die Definition eines Ähnlichkeitsmaßes für Trajektorien sich immer auf ihre Bestandteile, also Glieder, bezieht.

Die übliche Metrik, zwei Vektoren eines reellen Vektorraumes $\mathbb{R}^N$ mit N Dimensionen zu vergleichen, ist der *euklidische Abstand* (auch *euklidische Distanz*, engl. *euclidean distance*). Er definiert sich über die *L_2-Norm* des Distanzvektors der beiden zu vergleichenden Vektoren. Der euklidische Abstand kann als ein Spezialfall einer allgemeineren Abstandsfunktion verstanden werden.

Definition 5.1 (L_p-Norm) *Die von der **L_p-Norm** erzeugte Distanz zweier Vektoren $a = \langle a_i, \dots a_N \rangle$ und $b = \langle b_i, \dots b_N \rangle$ ist definiert als:*

$$L_p(a, b) = \Big(\sum_{i=1}^{N} |a_i - b_i|^p \Big)^{\frac{1}{p}}$$

Für $p = 1$ erhält man die **Manhattan-Distanz**:

$$L_1(a, b) = \sum_{i=1}^{N} |a_i - b_i|$$

Für $p = 2$ erhält man die **euklidische Distanz**:

$$L_2(a, b) = \sqrt{\sum_{i=1}^{N} |a_i - b_i|^2}$$

Für p gegen ∞ erhält man die von der Maximumsnorm abgeleitete Distanz:

$$L_\infty(a, b) = \lim_{p \to \infty} \left(\sum_{i=1}^{N} |a_i - b_i|^p \right)^{\frac{1}{p}} = \max_{i=1}^{N} |a_i - b_i|$$

Jede von einer L_p-Norm abgeleitete Distanz heißt auch **Minkowski-Metrik**. Diesen Begriff trennen wir von der Minkowski-Metrik, die sich auf die mit Einsteins Relativitätstheorie zusammenhängende Raumzeit bezieht. An dieser Stelle sei außerdem die **Minkowski-Summe** erwähnt, die sich als Menge von Summen der Vektoren zweier Teilmengen A und B eines Vektorraumes definiert: $A + B = \{a + b \mid a \in A, b \in B\}$.

Eine zusätzliche Abstraktion der L_p-Norm ist die gewichtete L_p-Norm. Bei ihr wird jeder Dimension ein Gewicht zugeordnet, sodass der Einfluss einer solchen in das Ergebnis flexibel ist.

Definition 5.2 (Gewichtete L_p-Norm) *Die von der **gewichteten** L_p**-Norm** erzeugte Distanz zweier Vektoren $a = \langle a_i, \ldots a_N \rangle$ und $b = \langle b_i, \ldots b_N \rangle$ mit der Gewichtung $w = \langle w_i, \ldots w_N \rangle$ ist definiert als:*

$$L_p{}'(a, b) = \left(\sum_{i=1}^{N} w_i \, |a_i - b_i|^p \right)^{\frac{1}{p}}$$

Für den Spezialfall $w_1 = w_2 = \ldots w_N$ erhalten wir die gewöhnliche L_p-Norm aus Definition 5.1. Die gewichtete Form ist inbesondere interessant, wenn es sich bei den Vektoren nicht um Primärdaten von Koordinaten, sondern um Vektoren eines Merkmalsraumes handelt, weil dort die Dimension eines bestimmten Merkmals wichtiger sein kann als eine andere. Die gewichtete L_p-Norm ist oft eine gute Lösung, wenn die gewöhnliche L_p-Norm den Anwendungsfall nicht zufriedenstellend bedienen kann [KP99, KCPM01].

5.2 Positions-Trajektorien-Vergleich

Einzelne Positionen

Üblicherweise vergleicht man eine Position p mit einer Trajektorie τ der Länge n, indem man erstere mit allen Positionen der Trajektorie vergleicht und diese Ergebnisse aggregiert. Das Aggregat der Wahl ist sinnvollerweise zumeist das Minimum [ZZ11, LS05].

Definition 5.3 (Positions-Trajektorien-Distanz) *Sei* d *ein Ähnlichkeitsmaß auf Positionen. Die Distanz zwischen einer Position* p *und einer Trajektorie* τ *definieren wir wie folgt:*

$$d_{\mathrm{PTD}}(p, \tau) = \min_{i=1}^{n} \big(d(p, \tau[i])\big)$$

Mehrere Positionen

Möchte man nicht nur eine Position mit einer Trajektorie vergleichen, sondern mehrere, ist es möglich, die Distanz für jede dieser Positionen zu berechnen und diese Werte schlicht zu summieren. Außerdem kann man die Menge von Positionen als Trajektorie auffassen und ein Ähnlichkeitsmaß zwischen Trajektorien bemühen. Beides ist aber oft nicht die optimale Lösung. Chen et al. schlagen in [CSZ+10] eine geschicktere Methode vor, mehrere Positionen mit einer Trajektorie zu vergleichen. Ihr Resultat heißt k-**Best-Connected Trajectory** (kBCT) und wird so motiviert:

> „A similarity function is normally designed by considering the actual application, and none of the similarity functions above satisfies the requirements of our applications, in which the query is broken into a small set of locations, and we concern more on whether a trajectory provides a good connection to query locations rather than whether the trajectory is similar to the query in shape. Therefore, due to the new application requirements, we need to define a new similarity function." [CSZ+10, Kap. 2]

Eine Anfrage besteht also aus einer Menge von Positionen $Q = \langle q_1, q_2, \ldots q_n \rangle$, die mit Trajektorien verglichen werden sollen, um diejenige zu finden, die diese Punkte am besten verbindet. Die Distanz zwischen einer einzelnen Position und einer Trajektorie wird genau wie in Definition 5.3 berechnet. Die Komposition des Unähnlichkeitsmaßes sieht dann wie folgt aus:

Definition 5.4 (k-Best-Connected Trajectory) *Sei* Q *eine Menge von Positionen und* τ *eine Trajektorie. Wie gut (beziehungsweise schlecht)* τ *die Positionen* Q *verbindet, messen wir wie folgt:*

$$s_{\mathrm{kBCT}}(Q,\tau) = \sum_{q \in Q} e^{-d_{\mathrm{PTD}}(q,\tau)}$$

wobei e *die Eulersche Zahl ist. Soll die Reihenfolge der Positionen relevant sein und ist* Q *eine geordnete Menge, sodass deren Positionen eine eindeutige Reihenfolge haben, definieren wir außerdem:*

$$s_{\mathrm{kBCT\_ord}}(Q,\tau) = \max \begin{cases} e^{-d_{\mathrm{PTD}}(\mathrm{head}(Q),\mathrm{head}(\tau))} + d_{\mathrm{kBCT\_ord}}(\mathrm{tail}(Q),\tau) \\ d_{\mathrm{kBCT\_ord}}(Q,\mathrm{tail}(\tau)) \end{cases}$$

Durch die Exponentialfunktion fließen Punkte mit kleiner Distanz stärker in das Ergebnis ein. Mit linear steigender Positionsdistanz nimmt der Summand exponentiell ab. Dadurch werden nur Trajektorien, die nah an allen Positionen von Q liegen, als ähnlich betrachtet, was der natürlichen Intuition für ein solches Ähnlichkeitsmaß am nächsten kommt [CSZ+10, Kap. 3].

5.3 Gewöhnlicher Trajektorienvergleich

Dieser Abschnitt bildet mit dem darauf folgenden den Kern der Liste von Unähnlichkeitsmaßen, denn nun geht es um die Berechnung der Ähnlichkeit unter Trajektorien. Jeder Ansatz wird zusammen mit seinem Ursprung – dazu gehören Autoren, Publikationsjahr und Motivation – vorgestellt. Eine chronologische Übersicht über diese Techniken und dazugehörige Ähnlichkeitsmaße bietet Tabelle 7.1 am Ende dieser Arbeit.

5.3.1 Closest-Pair-Distance

Eine einfache Möglichkeit, Trajektorien zu vergleichen, besteht darin, die Punktdistanzen aller möglichen Punktpaare zu berechnen und deren Minimum als Distanz der Trajektorien anzusehen.

Definition 5.5 (Closest-Pair-Distance) *Seien* σ *und* τ *Trajektorien der Länge* n *beziehungsweise* m *und* d *ein Unähnlichkeitsmaß auf Positionen. Dann bezeichnen wir mit der* ***Closest-Pair-Distance*** *(CPD) folgendes Unähnlichkeitsmaß auf Trajektorien:*

$$d_{\mathrm{CPD}}(\sigma,\tau) = \min_{i=1}^{n} \min_{j=1}^{m} \left(d(\sigma[i],\tau[j]) \right)$$

5.3.2 Aggregate über synchrone Glieder

Die Distanz zweier Trajektorien zu definieren, indem man jeweils die Distanzen der Glieder des gleichen Indexes berechnet und diese dann aggregiert, kann als *die* Standardmethode angesehen werden. Sie war auch die erste für den Vergleich von Zeitreihen vorgeschlagene Metrik [Che05, Kap. 2.2]. Ein **Aggregat** ist dabei das Ergebnis einer Funktion, die aus n Werten einen Wert erzeugt. Es gibt eine nicht zu unterschätzende Anzahl von Aggregatfunktionen. Für jedes Aggregat kann man ein dazugehöriges Ähnlichkeitsmaß definieren. Voraussetzungen für die Anwendung dieser Metrik sind kartesische Koordinaten als Positionen und identische Zeitstempel der zu vergleichenden Trajektorien. Wir bezeichnen solche Distanzen mit d_{AI}, weil die Positionen identischer Zeitstempel verglichen werden und darüber ein Aggegrat gebildet wird. Als Distanzfunktion der Punkte wird in aller Regel eine L_p-Norm bemüht, vorzugsweise die euklidische Distanz L_2.

Definition 5.6 (Aggregate über synchrone Glieder) *Seien σ und τ Trajektorien der Länge n und d eine Distanzfunktion auf Positionen. Wir definieren Unähnlichkeitsmaße für die folgenden relevanten Aggregate:*

- *Minimum:* $d_{AI\_min}(\sigma,\tau) = \min_{i=1}^{n}(d(\sigma[i],\tau[i]))$
- *Maximum:* $d_{AI\_max}(\sigma,\tau) = \max_{i=1}^{n}(d(\sigma[i],\tau[i]))$
- *Summe:* $d_{AI\_sum}(\sigma,\tau) = \sum_{i=1}^{n} d(\sigma[i],\tau[i])$
- *Arithmetisches Mittel:* $d_{AI\_mean}(\sigma,\tau) = \frac{1}{n} d_{sum}(\sigma,\tau) = \frac{1}{n}\sum_{i=1}^{n} d(\sigma[i],\tau[i])$
- *Quadratisches Mittel:* $d_{AI\_RMS}(\sigma,\tau) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} d(\sigma[i],\tau[i])^2}$
- *Median:*

$$d_{AI\_median}(\sigma,\tau) = \begin{cases} d(\sigma[\frac{n+1}{2}],\tau[\frac{n+1}{2}]) & , n \textit{ ungerade} \\ \frac{1}{2}\big(d(\sigma[\frac{n}{2}],\tau[\frac{n}{2}]) + d(\sigma[\frac{n}{2}+1],\tau[\frac{n}{2}+1])\big) & , n \textit{ gerade} \end{cases}$$

Es sind darüber hinaus weitere Aggregate denkbar, wie etwa das geometrische Mittel oder die Standardabweichung. An dieser Stelle sei darauf hingewiesen, dass $d_{CPD} \neq d_{AI\_min}$, weil bei letzterem nur Paarungen von Punkten mit gleichem Index berücksichtigt werden.

In der Literatur wird häufig von der euklidischen Distanz im Kontext von Trajektorien oder Zeitreihen gesprochen. Nicht immer wird spezifiziert, was genau damit gemeint ist. Das ist kritisch, weil der Begriff für unterschiedliche Varianten benutzt wird. Wir haben vier verschiedene Alternativen ausfindig gemacht. Agrawal et al., Faloutsos et al. und Morse et al. definieren sie als d_{AI_RMS} [AFS93, FRM94, MP07]. Yang et al., Zhang et al. und Needham et al. meinen mit der euklidischen Distanz d_{AI_mean} [YCW+12, ZHT06, NB03], Chen und Ng wiederum d_{AI_sum} [CN04b]. Bei eindimensionalen Zeitreihen ist

es außerdem üblich, sie als Vektoren der Dimension n zu handhaben und so direkt die euklidische Distanz oder eine andere L_p-Norm zu verwenden. Diese Methode kann auf mehrdimensionale Trajektorien übertragen werden. Eine solche Definition wird von Kim et al. und Keogh et al. verwendet [KPC01, KCPM01]. Verwendet man für die einzelnen Punktdistanzen ebenfalls die euklidische Distanz, erhält man ein Unähnlichkeitsmaß, das d_{AI_RMS} sehr ähnelt, sich aber davon durch die fehlende Normalisierung mit $\frac{1}{n}$ in der Wurzel unterscheidet. Dieses Unähnlichkeitsmaß bezeichnen wir mit d_E.

Definition 5.7 (Euklidische Distanz) *Seien σ und τ Trajektorien der Länge n und sei d eine Distanzfunktion auf Punkten.*

$$d_E(\sigma, \tau) = \sqrt{\sum_{i=1}^{n} d(\sigma[i], \tau[i])^2}$$

Im Laufe dieser Arbeit beziehen wir uns mit der euklidischen Distanz im Kontext von Trajektorien im engeren Sinne auf d_E und im weiteren Sinne auf alle vier Alternativen. In der Literatur ist es üblich, neue Metriken mit einer solchen Standardmethode zu vergleichen [DTS+08, CÖO05, CNOT07]. Weiming Hu et al. definieren in [HXF+07] eine Distanz, die genau d_{AI_mean} entspricht. Diese Distanz wird als **HU-Distanz** bezeichnet [MT09].

5.3.3 Aggregate über Positions-Trajektorien-Distanzen

Statt für ein Glied einer Trajektorie nur die Distanz zum Glied des gleichen Index mit identischem Zeitstempel der anderen Trajektorie zu berechnen und dies zu aggregieren, kann auch eine komplexere Berechnung stattfinden. Um genau zu sein, können wir die Distanz dieses Gliedes zu der gesamten anderen Trajektorie berechnen, wie in Definition 5.3. Solche Unähnlichkeitsmaße bezeichnen wir mit d_{AP}. Wie man sich leicht klarmachen kann, sind diese nicht symmetrisch. Das lässt sich jedoch leicht beheben, indem wir sie für beide Trajektorien zur jeweils anderen berechnen und den Mittelwert bilden. Genau wie zuvor können wir auf diese Weise Ähnlichkeitsmaße für beliebige Aggregate definieren. Der Einfachheit halber beschränken wir uns an dieser Stelle auf ein einzelnes Aggregat: das arithmetische Mittel.

Definition 5.8 (Aggregate über Positions-Trajektorien-Distanzen) *Seien* σ *und* τ *Trajektorien der Länge* n *beziehungsweise* m*. Wir definieren die Distanz zwischen* σ *und* τ *mittels PTD und dem arithmetischen Mittel als Aggregat wie folgt:*

$$d_{\mathrm{AP\_mean}}(\sigma,\tau) = \frac{1}{2}\left(d'_{\mathrm{AP\_mean}}(\sigma,\tau) + d'_{\mathrm{AP\_mean}}(\tau,\sigma)\right)$$

wobei

$$d'_{\mathrm{AP\_mean}}(\sigma,\tau) = \frac{1}{|\sigma|}\sum_{i=1}^{|\sigma|} d_{\mathrm{PTD}}(\sigma[i],\tau)$$

Dabei bezeichnet $|\sigma|$ *die Länge von* σ*, also* n *beziehungsweise* m*.*

Die nicht symmetrische Quasimetrik $d'_{\mathrm{AP\_mean}}$ wird auch **One-Way-Distance** (OWD) genannt [LS05, Abschn. 3.1].

5.3.4 Hausdorff-Distanz

Die *Hausdorff-Distanz* oder *Hausdorff-Metrik* ist ein in der Mathematik sehr verbreitetes Mittel zur Abstandsbestimmung und wurde vermutlich erstmals 1914 publiziert [Hau14]. Sie definiert eine Distanz zwischen zwei Teilmengen eines metrischen Raumes. Je mehr sich zwei Mengen überdecken, desto geringer ist ihre Hausdorff-Distanz.

Definition 5.9 (Hausdorff-Distanz) *Seien* S *und* T *zwei nicht-leere Teilmengen eines metrischen Raumes und* d *eine Distanzfunktion auf Punkten. Die* ***Hausdorff-Distanz*** *ist definiert als:*

$$d_{\mathrm{Hausdorff}}(S,T) = \max\{\, h(S,T), h(T,S) \,\}$$

wobei

$$h(S,T) = \sup_{t\in T}\inf_{s\in S} d(s,t)$$

Man könnte die Hausdorff-Distanz auch lediglich als h definieren und würde eine Quasimetrik erhalten, da diese nicht symmetrisch wäre. Die Maximumsfunktion und ihr zweites Argument mit vertauschten Mengen dienen ausschließlich dazu, die Funktion künstlich so zu verändern, dass sie die Eigenschaften einer Metrik erfüllt.

Da die Menge der Punkte $\{p_i \mid i \in \{1,\dots n\}\}$ einer Trajektorie $\tau = \big(\langle p_1,t_1\rangle, \langle p_2,t_2\rangle, \dots \langle p_n,t_n\rangle\big)$ als Teilmenge des Raumes, in dem sie liegen, verstanden werden können, kann man die Hausdorff-Distanz verwenden, um Trajektorien miteinander zu vergleichen [MT08, CÖO05, Che05, HKL05]. Wir können sie gewissermaßen als Gegenteil der Closest-Pair-Distance verstehen.

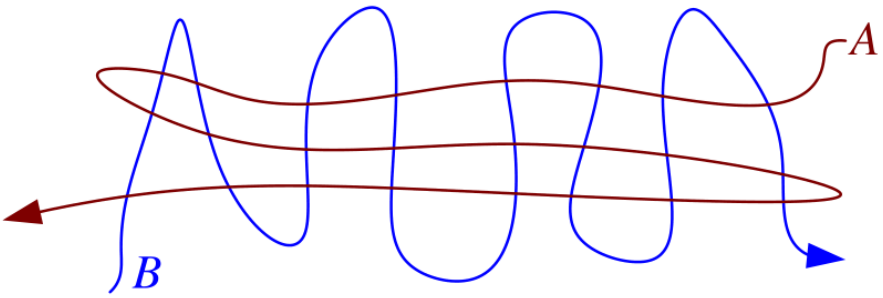


Abbildung 5.1: Zwei sich unähnliche Trajektorien, deren Hausdorff-Distanz trotzdem gering ist, da sie die Reihenfolge der Punkte ignoriert [AMP06].

Modifizierte Hausdorff-Distanz

Abbildung 5.1 verdeutlicht, dass beim Vergleich von Trajektorien oft gewünscht ist, dass die Reihenfolge der Punkte berücksichtigt wird. Vor diesem Hintergrund kamen Atev et al. auf die Idee, die Hausdorff-Distanz dahingehen zu modifizieren [AMP06].

Die Autoren machen zwei Schwächen der Hausdorff-Distanz aus. Einerseits ignoriert sie die Struktur der Mengen (Reihenfolge der Punkte), andererseits reagiert sie empfindlich auf Ausreißer.[1] Deswegen führen sie gezielte Änderungen an der Hausdorff-Distanz aus Definition 5.9 aus, sodass sie diese Schwächen nicht hat.

Definition 5.10 (Modifizierte Hausdorff-Distanz) *Seien* σ *und* τ *Trajektorien und* d *eine Distanzfunktion auf Punkten. Seien außerdem* $\alpha \in [0,1]$ *und* $N : P \to \mathcal{P}(P)$ *und* $C : P \to P$ *Funktionen. Wir definieren eine* ***Modifizierte Hausdorff-Distanz*** *(engl.* Modified Hausdorff Distance*) (MOHD) wie folgt:*

$$d_{\mathrm{MOHD}}(\sigma, \tau, \alpha, N, C) = \max\{\, h_\alpha(\sigma, \tau), h_\alpha(\tau, \sigma) \,\}$$

wobei

$$h_\alpha(\sigma, \tau) = \mathrm{ord}^{\alpha}_{s \in \sigma} \left(\min_{t \in N(C(s))} d(s, t) \right)$$

Dabei ist $N(t)$ *die Nachbarschaft von Punkt* t *innerhalb von* τ *und* $C(s)$ *der Punkt in* τ*, der sich relativ zur Länge der beiden Trajektorien an etwa gleichem Index wie* s *befindet.* $\mathrm{ord}^{\alpha}_{s \in \sigma} f(s)$ *bezeichnet den Wert von* $f(s)$*, der größer ist als* α *Prozent aller Werte* $f(s)$ *für die gesamte Trajektorie* σ*.*

Man beachte, dass die Schreibweise $s \in \sigma$ zwar die Glieder s der Trajektorie σ meint, jedoch den Fokus auf die Punkte legt und die Zeitstempel vernachlässigt. Mit α lässt sich eine Toleranz gegenüber Ausreißern einstellen. Je mehr er von 1 nach unten abweicht,

[1] Ausreißer werden in Abschnitt 6.10 Thema sein.

desto größer ist der Anteil der Punkte in der zweiten Trajektorie, der beim Vergleich ignoriert wird. Die Nachbarschaft $N(t)$ eines Punktes $t \in \tau$ wird sinnvollerweise schlicht als die Menge der Glieder von τ, deren Index um nicht mehr als einen festen Grenzwert vom Index von t abweicht, definiert. Damit soll das Verhalten der Hausdorff-Distanz auf Trajektorien wie in Abbildung 5.1 verhindert werden. Die Funktion C stellt sicher, dass die Trajektorien in korrekter Reihenfolge – also vom ersten bis zum letzten Glied – traversiert werden. Sie bewerkstelligt dies, indem der Punkt s, dessen Index β Prozent der Länge von σ entspricht, auf den Punkt in τ abgebildet wird, dessen Index ebenfalls β Prozent der Länge von τ entspricht. Abbildung 5.2 soll diese Funktionen versinnbildlichen. Die ursprüngliche Hausdorff-Distanz $d_{\text{Hausdorff}}$ ist ein Spezialfall der modifizierten d_{MOHD}, wenn $\alpha = 1$ und N und C so gewählt werden, dass $\forall\, s \in \sigma\colon N(C(s)) = \tau$.

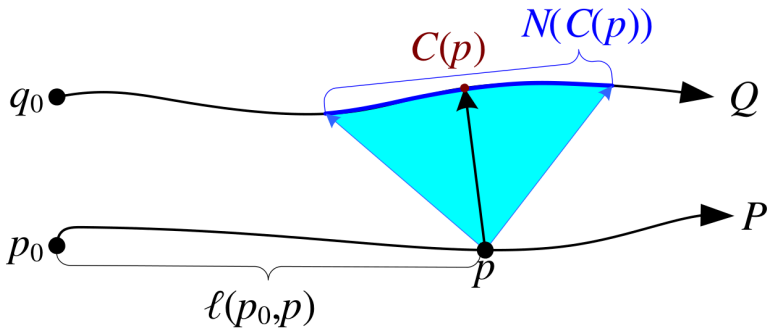


Abbildung 5.2: Bei der modifizierten Hausdorff-Distanz kann der Bereich für die Gliedpaarungen eingeschränkt werden [AMP06].

5.3.5 Fréchet-Distanz

Die *Fréchet-Distanz* ist eine Distanz zwischen Kurven [AG92]. Im Grunde ist eine Kurve nichts anderes als eine kontinuierliche Trajektorie mit dem Einheitsintervall als Urbildmenge. Wir definieren sie als Funktion $d_{\text{Fréchet}}(\sigma, \tau) : [0,1] \to S$, wobei S ein metrischer Raum mit Punktdistanzfunktion d ist. Dazu benötigen wir noch den Begriff der **Reparametrisierung** des Einheitsintervalls. Dies ist eine monoton steigende, surjektive Funktion, die ebenfalls in das Einheitsintervall abbildet. Die Menge aller möglichen Reparametrisierungen bezeichnen wir mit $\mathcal{R}$.

Definition 5.11 (Fréchet-Distanz) *Seien σ und τ Kurven in einem metrischen Raum. Die **Fréchet-Distanz** ist definiert als:*

$$d_{\text{Fréchet}}(\sigma, \tau) = \inf_{s \in \mathcal{R}} \inf_{t \in \mathcal{R}} \max_{i \in [0,1]} \{\, d\big(\sigma(s(i)), \tau(t(i))\big) \,\}$$

Die gängige intuitive Erklärung für die Fréchet-Distanz hat einen Hund und seinen Besitzer zum Gegenstand. Stellen wir uns einen Hundebesitzer vor, der seinen Hund an einer Leine spazieren führt. Die Kurven repräsentieren die Pfade, die beide auf ihrem Spaziergang jeweils zurücklegen. Zum Zeitpunkt i ist der Hund an Position $\sigma(s(i))$, während sein Besitzer an Position $\tau(t(i))$ ist. Wir berechnen die Distanz zwischen den beiden zu jedem Zeitpunkt und bilden darüber das Maximum, was der kürzesten möglichen Leine entspricht, die dafür nötig ist, den gesamten Spaziergang zu vollziehen. Das tun wir jedoch nicht für einen Fall, sondern für alle möglichen Parametrisierungen s und t, und versuchen dies zu minimieren. Das entspricht allen möglichen Gangarten von Hund und Besitzer: Sie können ihren Schritt verlangsamen oder beschleunigen, oder sogar stehen bleiben. Gesucht ist dann ein Paar von Gangarten, bei dem sie eine Leine möglichst geringer Länge brauchen.

Über die soeben definierte Fréchet-Distanz hinaus gibt es auch die **diskrete Fréchet-Distanz** mit dem Unterschied, dass die Kurve keine kontinuierliche Funktion, sondern eine diskrete ist, also aus endlich vielen (geradlinigen) Strecken besteht. Es werden dann nur deren Start- beziehungsweise Endpunkte berücksichtigt. Diese diskrete Fréchet-Distanz kann für diskrete Trajektorien benutzt werden, wobei die Start- und Endpunkte der Strecken die Glieder der Trajektorie sind.

Außerdem sei die **schwache Fréchet-Distanz** nicht unerwähnt. Bei ihr fällt in der Definition die Monotonie der Reparametrisierung weg: Sie darf also zu- und wieder abnehmen. Hund und Besitzer dürfen also sozusagen rückwärts laufen. Da Trajektorien per Definition aus monoton steigenden Zeitstempeln bestehen, kommt die Verwendung der schwachen Fréchet-Distanz für Trajektorien prinzipiell nicht in Frage.

Man kann die Fréchet-Distanz variieren, indem man statt der Maximumsfunktion andere Aggregate verwendet, zum Beispiel die Summe oder das arithmetische Mittel.

5.3.6 Dynamic-Time-Warping

Dynamic-Time-Warping ist eine weit verbreitete Technik zum Vergleich von Zeitreihen und Trajektorien ungleicher Länge. Sie erlaubt es, ein Glied einer Trajektorie mehrfach und mit einem zeitlich versetzten Glied einer zweiten Trajektorie zu vergleichen.

Ähnlichkeitsmaße, die es erlauben, ein Glied einer Trajektorie mit mehreren anderen Gliedern einer anderen Trajektorie zu vergleichen, bezeichnen wir als **elastisch**, andere als **unelastisch**, wie in Abschnitt 6.3 präziser definiert werden wird. Beispiele für unelastische Ähnlichkeitsmaße sind die Aggregate über synchrone Glieder (Abschnitt 5.3.2). DTW ist das erste Beispiel eines elastischen Ähnlichkeitsmaßes.

Definition 5.12 (Dynamic-Time-Warping) *Seien* σ *und* τ *Trajektorien der Länge* n *beziehungsweise* m *und* d *eine Distanzfunktion auf Punkten. Dann ist das mittels* ***Dynamic-Time-Warping*** *ermittelte Unähnlichkeitsmaß definiert als:*

$$d_{DTW}(\sigma,\tau) = \begin{cases} 0 & , n = 0 \wedge m = 0 \\ \infty & , n = 0 \vee m = 0 \\ d(head(\sigma), head(\tau)) + \min \begin{cases} d_{DTW}(tail(\sigma), tail(\tau)) \\ d_{DTW}(tail(\sigma), \tau) \\ d_{DTW}(\sigma, tail(\tau)) \end{cases} & , \textit{sonst} \end{cases}$$

Die Idee bei Dynamic-Time-Warping ist es, einen Punkt mehrfach mit Punkten der jeweils anderen Trajektorie paaren zu können, was bei unelastischen Ähnlichkeitsmaßen nicht möglich ist. Damit wird ein Punkt in der Zeit in gewissem Sinne verschoben. Diese Eigenschaft wird auch oft als *local time shifting* bezeichnet. Lokal ist sie deswegen, weil es sich nicht um eine globale Verschiebung handelt, sondern eine für jeden Punkt potentiell andere.

Üblicherweise wird Dynamic-Time-Warping in der Spracherkennung benutzt. Berndt und Clifford haben es in ihrer Pionierarbeit [BC94] 1994 erstmals für den Vergleich von Zeitreihen verwendet [DTS+08, Kap. 2].

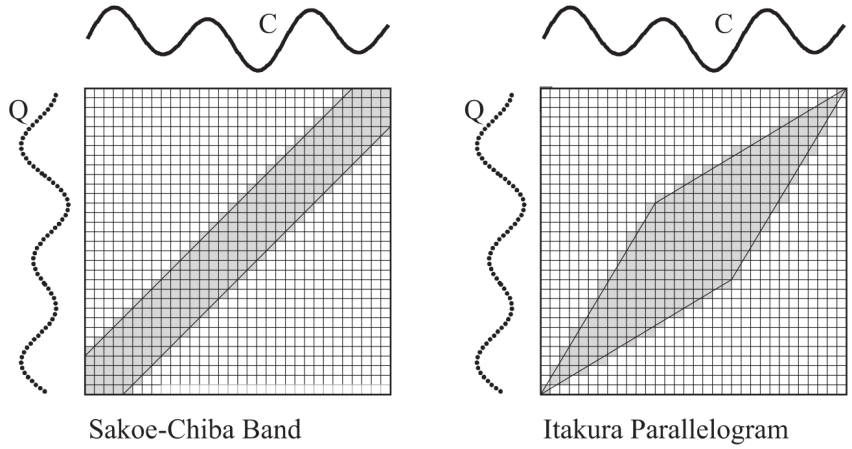


Abbildung 5.3: Ein Warping-Window beschränkt die erlaubten Gliedpaare [KR05].

Durch die rekursive und flexible allgemeine Definition von Dynamic-Time-Warping kann die Berechnung sehr teuer werden, wie wir in Abschnitt 6.8 genauer sehen werden. Um dem entgegenzuwirken, kommt häufig ein sogenanntes *warping window* zum Einsatz. Damit wird der Bereich, in dem ein Punkt verschoben werden darf, beschränkt. Genauer

gesagt, wird die Berechnung einzelner Gliedpaarungen bei einem Glied mit bestimmtem Index abgebrochen, sobald die Verschiebung zu dem Index des anderen Gliedes einen gewissen Grenzwert überschreitet. Man beachte, dass dieser Grenzwert nicht notwendigerweise konstant ist, sondern sich in Abhängigkeit des Index ändern kann. Die am häufigsten verwendeten *warping windows* sind das **Sakoe-Chiba-Band**, bei dem die mögliche Verschiebung konstant ist, und das **Itakura-Parallelogramm**, bei dem die Verschiebung umso größer sein darf, je weiter der Index von Start- und Endpunkt der Trajektorie entfernt ist. Abbildung 5.3 visualisiert die potentiellen Paare von Gliedern zweier Trajektorien Q und C. Auf den Achsen sind jeweils die Indizes der Glieder aufgetragen, die dunkleren Flächen bedeuten, dass die darin liegenden Punktpaare erlaubt sind, die in helleren sind es nicht.

Wir können die euklidische Distanz $\mathrm{d}_{\mathrm{sum}}$ als Spezialfall von $\mathrm{d}_{\mathrm{DTW}}$ verstehen, wenn wir das *warping window* auf konstant 1 limitieren [KR05].

5.3.7 Longest-Common-Subsequence

Unter dem Namen **Longest-Common-Subsequence** ist in der Informatik ein theoretisches Suchproblem bekannt, bei dem es darum geht, für zwei oder mehr gegebene Folgen ihre längste gemeinsame Teilfolge zu finden. Die längste gemeinsame Teilfolge der Folgen „ABBCADADBBD“ und „CBBABCCCADB“ ist zum Beispiel „ABCADB“. Vlachos et al. haben 2002 LCSS erstmals dafür verwendet, Unähnlichkeitsmaße auf Trajektorien zu definieren [VKG02]. Weil wir es statt mit einer endlichen Menge von diskreten Zeichen bei Positionen von Trajektorien zumeist mit reellwertigen Koordinaten – also einer weitaus größeren Obermenge – zu tun haben, ergibt es für die meisten Anwendungen keinen Sinn, auf exakte Gleichheit zu testen. Stattdessen wird eine Schranke definiert, bis zu der zwei Punkte gepaart werden dürfen, wenn ihre Distanz kleiner ist.

Die grundlegende Idee ist es – wie bei DTW – Punkte der ersten zu vergleichenden Trajektorie mit dem besten aus mehreren Punkten der zweiten Trajektorie zu paaren. Im Unterschied zu DTW dürfen dabei jedoch Punkte ausgelassen werden. Das Ergebnis der Berechnung ist nicht eine Summe der Distanzen von Punkten, sondern eine Anzahl von Punktpaaren, die gegebene Schranken nicht überschreiten. Dieses Konzept legt den Grundstein für eine neue Klasse von Ähnlichkeitsmaßen, wie wir in Abschnitt 6.5 genauer sehen werden.

Definition 5.13 (Longest-Common-Subsequence) *Seien* $\epsilon \in \mathbb{R}^+$ *und* $\delta \in \mathbb{N}$. *Sei außerdem* d *ein Unähnlichkeitsmaß auf Punkten. Dann ist* ***LCSS*** *für zwei Trajektorien* σ *und* τ *mit Länge* $\mathfrak{n}$ *beziehungsweise* $\mathfrak{m}$ *definiert als:*

$$\mathrm{LCSS}_{\epsilon,\delta}(\sigma,\tau) = \begin{cases} 0 & ,\mathfrak{n}=0 \vee \mathfrak{m}=0 \\ 1+\mathrm{LCSS}_{\epsilon,\delta}(\mathrm{tail}(\sigma),\mathrm{tail}(\tau)) & ,\mathrm{d}(\mathrm{head}(\sigma),\mathrm{head}(\tau))<\epsilon \wedge |\mathfrak{n}-\mathfrak{m}|<\delta \\ \max\begin{cases}\mathrm{LCSS}_{\epsilon,\delta}(\mathrm{tail}(\sigma),\tau) \\ \mathrm{LCSS}_{\epsilon,\delta}(\sigma,\mathrm{tail}(\tau))\end{cases} & ,\textit{sonst} \end{cases}$$

Zwei Punkte werden gepaart und inkrementieren das Ergebnis, wenn ihre Distanz kleiner als ϵ ist und ihre Indizes weniger als δ beieinander liegen. Der mittels LCSS gewonnene Wert ist umso höher, je mehr Punkte gepaart werden können. Die Schranke ϵ wird häufig Paarungsschwelle (engl. *matching threshold*) genannt. Auch LCSS ist elastisch, sogar in einem höheren Maße, weil gleichsam Lücken in den Gliedern erlaubt werden, die nicht gepaart werden, ohne dass die Ähnlichkeit der Trajektorien nicht mehr erkannt wird.[2] Dies und die Tatsache, dass nicht der tatsächliche Abstand der Punkte in das Ergebnis einfließt, sondern nur die quantisierten Werte 1 und 0, macht LCSS im Vergleich zu den bisherigen Ähnlichkeitsmaßen wie den euklidischen Distanzen oder DTW sehr robust gegen Rauschen und Ausreißer:

> „LCSS measure matches two sequences by allowing them to stretch, without rearranging, the sequence of the elements, but allowing some elements to be unmatched (which is the main advantage of the LCSS measure compared with Euclidean Distance and DTW). Therefore, LCSS can efficiently handle outliers and different scaling factors." [FGT07, Kap. 2]

In der Originalpublikation [VKG02] wird die Schranke ϵ für die zwei kartesischen Raumdimensionen einzeln berechnet. Das ähnelt der Manhattan-Distanz für die Punkte, da jede Dimension einzeln betrachtet wird, unterscheidet sich aber von ihr, weil nicht die Summe der beiden Teildistanzen entscheidend ist.

Ausgehend von LCSS definieren wir zwei Ähnlichkeitsmaße $\mathrm{d}_{\mathrm{LCSS}}$ relativ zur Länge der Trajektorien, die über die Trajektorien σ und τ hinaus als Parameter die Schranken ϵ und δ konsumieren. Das zweite dieser Ähnlichkeitsmaße bedient sich des ersten, erlaubt aber beliebige Translationen $\oplus$ im Raum der Trajektorien $\mathbb{S}$, um das Ergebnis zu maximieren.

[2] Genaueres dazu in Abschnitt 6.3.

Definition 5.14 (LCSS-Ähnlichkeitsmaß)

$$s_{LCSS\_1}(\sigma, \tau, \epsilon, \delta) = \frac{LCSS_{\epsilon,\delta}(\sigma, \tau)}{\min(\mathfrak{n}, \mathfrak{m})}$$

$$s_{LCSS\_2}(\sigma, \tau, \epsilon, \delta) = \max_{a \in S}(s_{LCSS\_1}(\sigma, \tau \oplus a, \epsilon, \delta))$$

Aus diesen beiden Ähnlichkeitsmaßen definieren wir schließlich Unähnlichkeitsmaße. Dabei kommt es gelegen, dass s_{LCSS_1} per Definition nicht größer als 1 sein kann.

Definition 5.15 (LCSS-Unähnlichkeitsmaß)

$$d_{LCSS\_1}(\sigma, \tau, \epsilon, \delta) = 1 - s_{LCSS\_1}(\sigma, \tau, \epsilon, \delta)$$

$$d_{LCSS\_2}(\sigma, \tau, \epsilon, \delta) = 1 - s_{LCSS\_2}(\sigma, \tau, \epsilon, \delta)$$

In [BYÖ97] definieren Bozkaya et al. schon 1997 ein Ähnlichkeitsmaß, das äquivalent zu LCSS ist [MdB02].

Sigmoidfunktion statt fester Paarungsschwelle

Vlachos et al. schlagen in einer weiteren Publikation vor, ihre LCSS-Technik mit einer Sigmoidfunktion zu verbessern [VGK02].

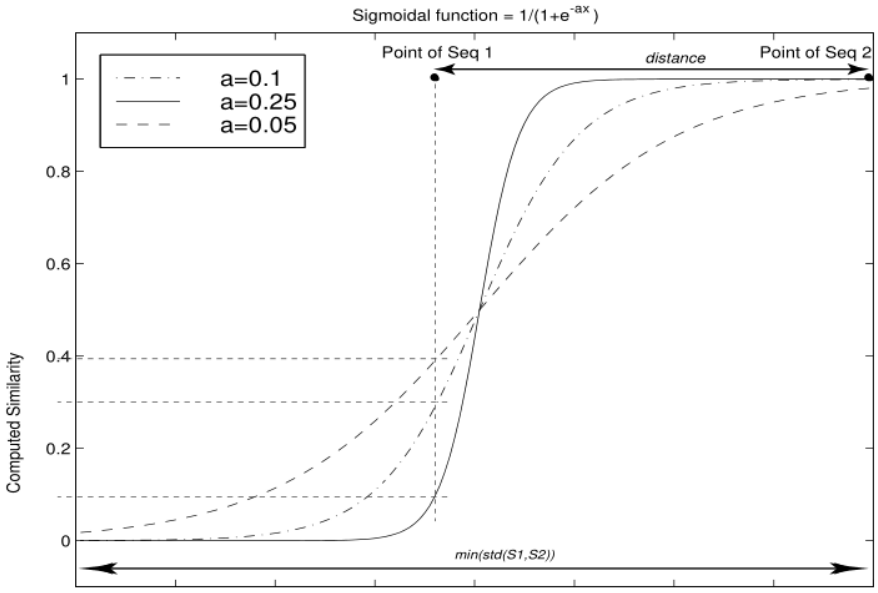


Abbildung 5.4: Sigmoidfunktion [VGK02].

Das Problem bei LCSS sei, dass Punktpaare, deren Distanz nur knapp über der Paarungsschwelle ϵ liegt, genauso hart bestraft werden, wie Punktpaare, die sehr weit auseinander liegen. Das mache die Wahl einer guten Paarungsschwelle zu einem kritischen Problem. Die Verbesserung besteht darin, sie durch eine gewichtete Paarungsfunktion zu ersetzen, die tatsächlich von der Distanz der Punkte abhängt. Diese Funktion ist eine Sigmoidfunktion, die also eine „S“-Form hat und sich durch Parameter anpassen lässt, wie man in Abbildung 5.4 sieht.

Definition 5.16 (LCSS mit einer Sigmoidfunktion) *Seien* $a \in \mathbb{R}, k \in \mathbb{N}$ *und* $\delta \in \mathbb{N}$. *Dann ist* ***LCSS mit einer Sigmoidfunktion*** *für zwei Trajektorien* σ *und* τ *mit Länge* n *beziehungsweise* m *definiert als:*

$$\begin{aligned}\text{SigmoidSim}_{a,k,\delta}(\sigma,\tau) = \text{SigmoidMatch}_{a,k}(\text{head}(\sigma),\text{head}(\tau)) + \\ \max\begin{cases}\text{SigmoidSim}_{a,k,\delta}(\text{tail}(\sigma),\text{tail}(\tau)) \\ \text{SigmoidSim}_{a,k,\delta}(\text{tail}(\sigma),\tau) & ,|n-m|<\delta \\ \text{SigmoidSim}_{a,k,\delta}(\sigma,\text{tail}(\tau))\end{cases}\end{aligned}$$

wobei

$$\text{SigmoidMatch}_{a,k}(p_\sigma,p_\tau) = \begin{cases}0 & ,L_1(p_\sigma,p_\tau) > \min(\text{std}(\sigma,\tau)) \\ s(\lceil \frac{L_1(p_\sigma,p_\tau)}{\min(\text{std}(\sigma,\tau))} \cdot (2k+1)\rceil) & ,\textit{sonst}\end{cases}$$

und $s(x) = \frac{1}{1+e^{-a(x-k-1)}}$ *für* $x \in \{1,\dots 2k+1\}$ *die Sigmoidfunktion ist.*

Der Parameter δ ist uns aus Definition 5.13 bekannt. a und k geben an, wie sehr die Sigmoidfunktion geneigt ist beziehungsweise wie genau die Paarung stattfinden soll.

Genau wie aus LCSS können wir aus SigmoidSim Ähnlichkeits- und Unähnlichkeitsmaße mit dem Einheitsintervall als Wertebereich ableiten: In Definition 5.14 wird lediglich LCSS durch SigmoidSim ersetzt. Wir bezeichnen die so entstehenden Ähnlichkeitsmaße mit $s_{\text{LCSS\_SM\_1}}$ und $s_{\text{LCSS\_SM\_2}}$ und die dazugehörigen Unähnlichkeitsmaße mit $d_{\text{LCSS\_SM\_1}}$ und $d_{\text{LCSS\_SM\_2}}$.

5.3.8 Edit-Distance-on-Real-Sequence

Lei Chen schlägt 2005 zwei weitere Unähnlichkeitsmaße vor, die versuchen, LCSS beziehungsweise DTW weiter zu verbessern: *Edit Distance on Real Sequence* (EDR) [CÖO05] und *Edit Distance with Real Penalty* (ERP) [CN04b].

EDR bedient sich dabei der insbesondere in der Bioinformatik und Spracherkennung verbreiteten *Edit Distance* von Vladimir Levenshtein [Lev66]. Ihr Ansatz ist, die Ähnlichkeit von zwei Sequenzen zu messen, indem die Anzahl der Einfüge-, Lösch- oder Ersetzungsaktionen bestimmt wird, die mindestens dafür notwendig sind, die erste Sequenz in die zweite umzuwandeln. Chen verwendet diese Technik nun für Trajektorien. Genau wie bei LCSS wird Gleichheit von Positionen hierbei nicht als exakte Übereinstimmung, sondern als Unterschreitung eines bestimmten Grenzwertes ihrer Distanz definiert.

Definition 5.17 (Edit-Distance-on-Real-Sequence) *Seien σ und τ Trajektorien der Länge n beziehungsweise m, $\epsilon \in \mathbb{R}^+$ eine Schranke und d ein Unähnlichkeitsmaß auf Punkten. Dann definieren wir die* ***Edit-Distance-on-Real-Sequence*** *durch:*

$$d_{\mathrm{EDR}}(\sigma, \tau, \epsilon) = \begin{cases} n & , m = 0 \\ m & , n = 0 \\ \min \begin{cases} \mathrm{subcost} + d_{\mathrm{EDR}}(\mathrm{tail}(\sigma), \mathrm{tail}(\tau), \epsilon) \\ 1 + d_{\mathrm{EDR}}(\mathrm{tail}(\sigma), \tau, \epsilon) \\ 1 + d_{\mathrm{EDR}}(\sigma, \mathrm{tail}(\tau), \epsilon) \end{cases} & , \textit{sonst} \end{cases}$$

wobei

$$\mathrm{subcost} = \begin{cases} 0 & , d(\mathrm{head}(\sigma), \mathrm{head}(\tau)) \leqslant \epsilon \\ 1 & , \textit{sonst} \end{cases}$$

Auch hier werden in der Originaldefinition die Differenzen der Koordinaten der Punkte einzeln berechnet und mit ϵ verglichen.

Die Definition liefert ein Unähnlichkeitsmaß, das also einen kleineren Wert annimmt, je ähnlicher sich zwei Trajektorien sind. Wie bei LCSS werden Abstände im Ergebnis mit 1 und 0 quantisiert, sodass Ausreißer das Ergebnis nicht massiv verfälschen können. Der wesentliche Unterschied besteht jedoch darin, dass nicht gepaarte Punkte nicht wie bei LCSS ignoriert werden, sondern den Wert erhöhen und Lücken somit bestrafen. Das mache d_{EDR} akkurater als $d_{\mathrm{LCSS\_1}}$:

> „Contrary to LCSS, EDR assigns penalties to the gaps between two matched subtrajectories according to the lengths of gaps, which makes it more accurate than LCSS.“ [CÖO05, Abschn. 3.1]

Im Gegensatz zu LCSS sind werden die unähnlichen Subtrajektorien zweier Trajektorien nicht ignoriert, weil deren Längen einfließen. Tatsächlich könnten wir EDR für zwei Trajektorien der Länge n beziehungsweise m auch wie folgt definieren: $d_{\mathrm{EDR}}(\sigma, \tau, \epsilon) = n + m - 2 \cdot \mathrm{LCSS}_{\epsilon,\infty}(\sigma, \tau)$.

5.3.9 Edit-Distance-with-Real-Penalty

In Abschnitt 6.7 werden wir sehen, dass EDR die Eigenschaften einer Metrik nicht erfüllt. Alle bisherigen elastischen Unähnlichkeitsmaße, namentlich d_{DTW}, d_{LCSS} und d_{EDR} tun das nicht, unterstützen aber *local time shifting*. Das unterscheidet sie von der Familie der von L_p-Normen abgeleiteten Unähnlichkeitsmaße, die zwar metrisch sind, aber *local time shifting* nicht unterstützen. Dies motiviert den Entwurf von ERP. Chen verbindet in seiner viel beachteten Publikation [CN04b] erstmals beide Eigenschaften in einer Metrik.

Definition 5.18 (Edit-Distance-with-Real-Penalty) *Seien σ und τ Trajektorien der Länge n beziehungsweise m. Seien außerdem g eine zufällige Position und d eine Distanzfunktion auf Positionen. Dann definieren wir die* ***Edit-Distance-with-Real-Penalty*** *durch:*

$$d_{ERP}(\sigma,\tau) = \begin{cases} \sum_{i=1}^{n} d(\sigma[i], g) & , m = 0 \\ \sum_{i=1}^{m} d(\tau[i], g) & , n = 0 \\ \min \begin{cases} d(\text{head}(\sigma), \text{head}(\tau)) + d_{ERP}(\text{tail}(\sigma), \text{tail}(\tau)) \\ d(\text{head}(\sigma), g) + d_{ERP}(\text{tail}(\sigma), \tau) \\ d(\text{head}(\tau), g) + d_{ERP}(\sigma, \text{tail}(\tau)) \end{cases} & , \textit{sonst} \end{cases}$$

Man sieht deutlich die Parallelen zu EDR. Allerdings wird bei ERP der Parameter ϵ vermieden, da er für die Verletzung der Dreiecksungleichung verantwortlich ist. Stattdessen kommt eine zufällige Position g zum Einsatz. Sie dient lediglich dazu, dass tatsächlich Distanzen aus dem Raum der Trajektorien in das Ergebnis einfließen und keine Quantisierung stattfindet oder Glieder ungepaart bleiben können. So bleibt die Erfüllung der Dreiecksungleichung gewahrt. Tatsächlich spielt es keine Rolle, wie g gewählt ist, solange die Position konstant ist [CN04b, Abschn. 3.2].

5.3.10 Sequence-Weighted-Alignment-Model

Im Jahr 2007 publizieren Morse und Patel einen Ansatz, der zwei neue Ideen beinhaltet [MP07]. Zum einen handelt es sich um **Fast-Time-Series-Evaluation** (FTSE), einen Algorithmus, der verwendet werden kann, um verschiedene Distanzmaße effizient auszuwerten. Das betrifft zum Beispiel LCSS und EDR. Der Algorithmus hat nach Angaben der Autoren Vorzüge gegenüber anderen Strategien, und zwar sowohl gegenüber dynamischer Programmierung als auch gegenüber Strategien mit einem *warping window*. Zum anderen schlagen sie das Ähnlichkeitsmodell **Sequence-Weighted-Alignment-Model** (Swale)

vor, das in gewisser Weise eine Abstraktion von LCSS und EDR darstellt. Genau wie LCSS und EDR profitiert es von FTSE. Der Entwurf von Swale gründet sich prinzipiell auf der Vermeidung der Schwächen von LCSS und EDR. Beide bilden Gliedpaare je nach Ähnlichkeit ihrer Positionen. Der Wert für ein auf LCSS basierendes Unähnlichkeitsmaß ergibt sich dadurch, dass Ähnlichkeit von Gliedern belohnt wird. Deren Unähnlichkeit wird allerdings nicht bestraft, egal wie viele Glieder sich unähnlich sind. In Abbildung 5.5 ist $\mathrm{LCSS}(C, C) = \mathrm{LCSS}(C, D)$, was offensichtlich nicht akkurat ist. EDR hingegen berechnet sich dadurch, dass es die Unähnlichkeit der Glieder bestraft, ihre Ähnlichkeit aber ignoriert. In Abbildung 5.5 ist $d_{\mathrm{EDR}}(A, B) = d_{\mathrm{EDR}}(C, D)$, obwohl C und D sich intuitiv ähnlicher sind.

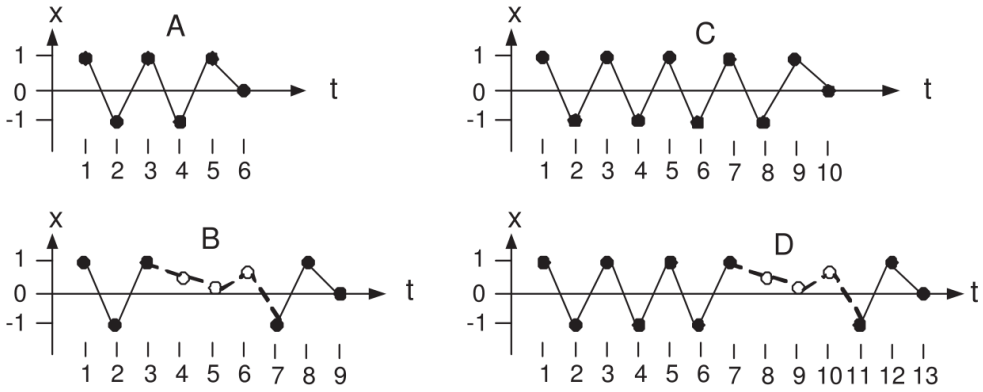


Abbildung 5.5: Beispielhafte Zeitreihen für LCSS, EDR und Swale [MP07].

Swale erlaubt es, sowohl die Belohnung bei erfolgreichen Paarungen als auch die Bestrafung bei nicht erfolgreichen Paarungen über Parameter einzustellen. Beide Parameter können darüber hinaus relativ zueinander gewichtet werden.

Definition 5.19 (Sequence-Weighted-Alignment-Model) *Seien σ und τ Trajektorien der Länge n beziehungsweise m. Seien außerdem d ein Unähnlichkeitsmaß auf Punkten, ϵ eine Paarungsschwelle, ζ der Wert der Bestrafung einer Lücke und η der Wert der Belohnung einer Paarung. Dann definieren wir das Ähnlichkeitsmaß nach **Swale** durch:*

$$s_{\mathrm{Swale}}(\sigma, \tau, \epsilon, \zeta, \eta) = \begin{cases} n \cdot \zeta & , m = 0 \\ m \cdot \zeta & , n = 0 \\ \eta + s_{\mathrm{Swale}}(\mathrm{tail}(\sigma), \mathrm{tail}(\tau), \epsilon, \zeta, \eta) & , d(\mathrm{head}(\sigma), \mathrm{head}(\tau)) < \epsilon \\ \max \begin{cases} \zeta + s_{\mathrm{Swale}}(\mathrm{tail}(\sigma), \tau, \epsilon, \zeta, \eta) \\ \zeta + s_{\mathrm{Swale}}(\sigma, \mathrm{tail}(\tau), \epsilon, \zeta, \eta) \end{cases} & , \textit{sonst} \end{cases}$$

Hier benennen wir die Funktion tatsächlich mit s statt d, weil sie ein Ähnlichkeitsmaß im engeren Sinne ist, also ähnliche Trajektorien tatsächlich größere Werte nach sich ziehen. Wir sehen deutlich die Parallelen zu LCSS und EDR. Die Parameter ζ und η erlauben es hier jedoch, den jeweiligen Einfluss der beiden Konzepte genau zu bestimmen. Für ein sinnvolles Ähnlichkeitsmaß sollten Belegungen gewählt werden, bei denen $\zeta \leqslant 0$ und $\eta \geqslant 0$ ist. Die Gewichtung lässt sich an dem Verhältnis $\frac{\zeta}{\eta}$ messen. Wie bei LCSS und EDR werden die Differenzen der Koordinaten der Punkte in der Originalpublikation einzeln berechnet und mit ϵ verglichen. Im Übrigen gehen die Autoren von wie in Abschnitt 4.2.1 umschrieben normalisierten Trajektorien aus.

Durch die zuvor beschriebenen Schwächen gibt es Datensätze, auf denen die Verwendung von LCSS bessere Ergebnisse erzielt als die von EDR, und solche, bei denen es sich umgekehrt verhält. Experimente zeigen, dass Swale auf *beiden* Arten von Datensätzen den anderen beiden Ähnlichkeitsmaßen noch überlegen ist, sowohl auf für LCSS als auch auf für EDR besser geeigneten [MP07, Abschn. 6.3].

5.3.11 Piciarelli-Foresti-Distanz

Piciarelli und Foresti haben festgestellt, dass bei bewegten Objekten mit kleinen Geschwindigkeitsunterschieden, die eigentlich sehr ähnliche Trajektorien haben, ein *time drift* entsteht, sodass ihre euklidische Distanz sehr groß ist. In Abbildung 5.6 ist ein ein solcher Fall dargestellt. Die Alternativen DTW und LCSS genügen ihren Anforderungen nicht, da ihre Anwendung im Bereich des Online-Clustering liegt.[3] Sie haben daraufhin eine neue Metrik entwickelt, bei der für jedes weitere Glied das Zeitfenster für potentielle Gliedpaare größer wird [PF06].

Definition 5.20 (PF-Distanz) *Seien σ und τ Trajektorien der Länge n beziehungsweise m. Sei außerdem $\delta \in \mathbb{R}$ eine Konstante zur Einstellung des Bereiches, in dem eine zeitliche Verschiebung stattfinden darf. Die Piciarelli-Foresti-Distanz (**PF-Distanz**) ist dann definiert als:*

$$d_{\mathrm{PF}}(\sigma, \tau, \delta) = \frac{1}{n} \sum_{i=1}^{n} d'_{\mathrm{PF}}(\sigma[i], \tau)$$

wobei

$$d'_{\mathrm{PF}}(\sigma[i], \tau) = \min_{j} \left(L_2(\sigma[i], \tau[j])\right) \quad \textit{mit} \quad j \in \{\lfloor (1-\delta)i \rfloor \ldots \lceil (1+\delta)i \rceil\}$$

[3] Die genauen Gründe hierfür werden wir in Kapitel 6 beleuchten.

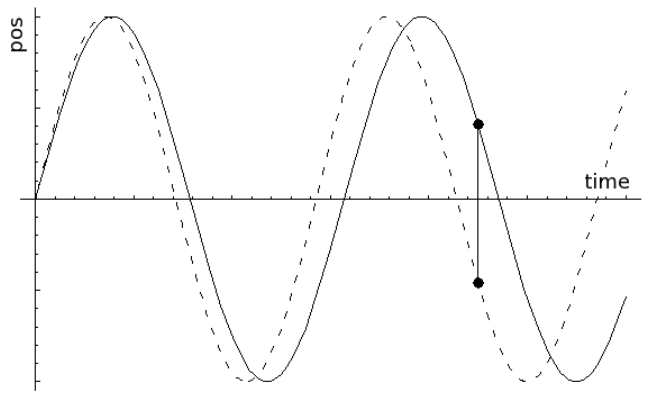


Abbildung 5.6: Große euklidische Distanz für ähnliche Trajektorien [PF06].

Dadurch, dass die Menge, aus der `j` ausgewählt wird, von `i` abhängt, wird sie bei zunehmender Zeit immer größer, sodass mehr Punkte der zweiten Trajektorie mit einem Punkt der ersten gepaart werden dürfen.

Die Originaldefinition vergleicht Trajektorien mit Clustern statt mit anderen Trajektorien. Dabei wird das Argument der Minimumsfunktion zusätzlich mit einem Parameter normalisiert, was für uns nicht relevant ist, sodass wir die Formel vereinfachen. Diese Anpassung wird auch in [MT09] vollzogen.

5.3.12 Shape-Based-Distance

Yanagisawa et al. schlagen 2003 eine neue Methode für die Indizierung von Trajektorien vor, die den Fokus auf deren Form (engl. *shape*) legt [YAS03]. Dabei werden statt wie üblich Punkte der Trajektorien nun gewissermaßen die Strecken zwischen ihnen verglichen, um die Ähnlichkeit zu bestimmen:

> „[. . .] our approach is based on the shape similarity between lines, while the existing approaches adopt the distance between points as the key to retrieve required objects.“ [YAS03, Kap. 1]

Wir nennen diese Technik **Shape-Based-Distance** (SBD). Sie orientiert sich stark an der euklidischen Distanz [FGT07, Kap. 2]. Die Autoren unterscheiden drei Arten der Ähnlichkeit: spatiotemporale Ähnlichkeit, räumliche Ähnlichkeit und zeitliche Ähnlichkeit. Unter letzterer verstehen sie die übliche euklidische Distanz und definieren sie folglich nicht explizit. Für die ersten beiden liefern sie eine erstaunlich einfache Definition, die im Grunde ebenfalls einer mit der Länge der Trajektorien normalisierten euklidischen Distanz entspricht. Ihre entscheidende Idee ist jedoch ein vorgeschaltetes **Resampling** der

Trajektorien, sodass die Berechnung nicht auf den ursprünglichen Primärdaten, sondern einem neuen Sampling der kontinuierlichen Trajektorie durchgeführt wird. Damit befinden wir uns genau im Spannungsfeld zwischen kontinuierlichen Trajektorien und deren diskreter Darstellung. Das Resampling der Trajektorien sorgt für deren Normalisierung in zeitlicher oder räumlicher Dimension. Man greift beim Resampling auf die Methoden der Interpolation zurück, die wir in Abschnitt 4.2.2 kennengelernt haben; um genau zu sein, auf PLA.

Definition 5.21 (normalisierte Trajektorie) *Sei τ eine im Zeitintervall $[t_S, t_E]$ definierte Trajektorie. Die zu τ gehörende **zeitlich normalisierte Trajektorie** der Länge m ist definiert als:*

$$\tau_{\Delta t} = \big(\langle\tau(t_S), t_S\rangle, \langle\tau(t_S + \Delta t), t_S + \Delta t\rangle, \ldots \langle\tau(t_S + m\Delta t), t_S + m\Delta t\rangle\big)$$

*wobei $t_S + m\Delta t = t_E$. Die zu τ gehörende **räumlich normalisierte Trajektorie** mit Streckenlänge δ ist definiert als:*

$$\tau_\delta = \big(\langle\tau(t_S), t_S\rangle, \ldots \langle\tau(t_i), t_i\rangle, \ldots \langle\tau(t_E), t_E\rangle\big)$$

wobei $d(\tau_\delta[t_i], \tau_\delta[t_{i+1}]) = \delta$ für alle $t_i \in (t_s \ldots t_E)$.

Eine zeitlich normalisierte Trajektorie hat also äquidistante Zeitstempel, während eine räumlich normalisierte Trajektorie Strecken der gleichen Länge hat. Mithilfe dieser normalisierten Trajektorien können wir nun eine Minkowski-Metrik definieren:

Definition 5.22 (Shape-Based-Distance) *Seien σ und τ Trajektorien. Dann seien die zeitlich oder räumlich normalisierten Trajektorien dazu σ' beziehungsweise τ' mit der Länge m. Die Shape-Based-Distance (**SBD**) ist wie folgt definiert:*

$$d_{SBD}(\sigma, \tau) = \frac{1}{m}\, d_E(\sigma', \tau') = \frac{1}{m}\sqrt{\sum_{i=1}^{m} L_2(\sigma'[i], \tau'[i])^2}$$

Man beachte, dass die zu vergleichenden Trajektorien, wenn sie räumlich normalisiert werden, die gleiche Verschiebung haben müssen (Definition 4.5). Bei Verwendung der zeitlichen Normalisierung erhalten wir die spatiotemporale Distanz d_{SBD_TS}, bei räumlicher Normalisierung die räumliche Distanz d_{SBD_S}.

Nehmen wir zur Verdeutlichung zwei Trajektorien $\sigma = \big(\langle(1,7),1\rangle, \langle(3,7),2\rangle, \langle(3,2),3\rangle, \langle(4,1),7\rangle\big)$ und $\tau = \big(\langle(1,7),1\rangle, \langle(3,7),2\rangle, \langle(3,2),5\rangle, \langle(4,1),7\rangle\big)$ an, die zwar den gleichen Pfad zurücklegen, aber zu versetzten Zeiten. Während die euklidische Distanz zwischen

ihnen 0 ist, können wir mit d_{SBD_TS} eine tatsächliche Distanz > 0 feststellen. Ebenso können wir mit d_{SBD_S} Trajektorien der gleichen Form unabhängig von ihren Zeitkomponenten finden.

5.3.13 Area-Based-Distance

Im Jahr 2003 veröffentlichen Needham und Boyle eine Publikation, in der sie ein Unähnlichkeitsmaß auf Trajektorien vorschlagen, das der Fläche zwischen ihnen entspricht. Sie nutzen es dafür, die Genauigkeit von positionalen Trackern zu messen. Wir nennen es **Area-Based-Distance** (ABD) [NB03].

Es wird ein zweidimensionaler euklidischer Raum angenommen. Für die Distanz von Punkten wird wie üblich die euklidische Distanz L_2 verwendet. Die Autoren berücksichtigen, dass zwei sich sonst unähnliche Trajektorien sich ähnlich sein können, wenn man von einer Verschiebung absieht. Darunter fallen räumliche Verschiebung (Translation), zeitliche Verschiebung sowie spatiotemporale Verschiebung. Die optimale Translation $\hat{d}$ einer Trajektorie τ, um die Distanz zu einer anderen Trajektorie σ zu minimieren, wird zum Beispiel als durchschnittliche Distanz aller ihrer Glieder berechnet, und zwar pro Dimension: $\hat{d}_x = \frac{1}{n}\sum_{i=1}^{n}(\sigma[i]_x - \tau[i]_x)$, $\hat{d}_y = \frac{1}{n}\sum_{i=1}^{n}(\sigma[i]_y - \tau[i]_y)$, wobei n die Länge beider Trajektorien ist. Führt man eine Translation von τ um $\hat{d}$ durch, so wird stets gelten: $d_{mean}(\sigma, \tau \oplus \hat{d}) \leqslant d_{mean}(\sigma, \tau)$. Weiterhin definieren Needham und Boyle auch optimale zeitliche und spatiotemporale Verschiebungen. Diese Verschiebungen werden gegebenenfalls vor der Berechnung der Distanz durchgeführt.

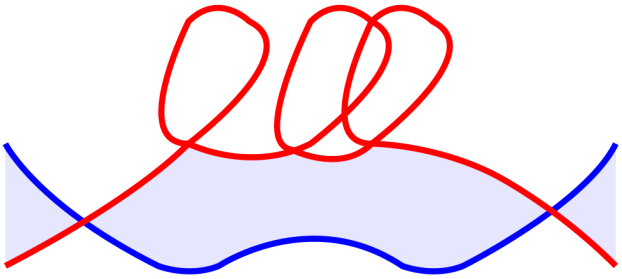


Abbildung 5.7: Die Fläche zwischen Trajektorien misst deren Unähnlichkeit [NB03].

Die Distanz zwischen zwei Trajektorien soll nun die Fläche zwischen ihnen sein. Zu diesem Zweck werden zunächst alle Schnittpunkte miteinander berechnet, um die Regionen zu bestimmen, deren Flächen in der Summe die gesamte Fläche ergeben. Schneidet eine Trajektorie sich selbst, wird dieser so erzeugte Zyklus entfernt. Eine Region ist ein n-seitiges Polygon, das entlang seiner Kanten – das sind die Strecken der Trajektorien, die

an dieser Region teilhaben – durchlaufen wird. Die Fläche zwischen jeder Kante und der Abszissenachse wird durch $\frac{(x_{i+1}-x_i)(y_i-y_{i+1})}{2}$ berechnet und je nach Vorzeichen zu dem Ergebnis addiert oder subtrahiert, sodass nur positive Flächen Summanden sind. Nach Umformungen ergibt sich folgende Formel:

Definition 5.23 (Area-Based-Distance) *Seien σ und τ Trajektorien. Dann bezeichnen wir mit R die Menge der Regionen zwischen σ und τ. Die Area-Based-Distance (**ABD**) definieren wir als:*

$$d_{ABD}(\sigma,\tau) = \sum_{r\in R} |A(r)|$$

wobei die Fläche einer Region sich wie folgt berechnet:

$$A(r) = \frac{1}{2}\left(\left(\sum_{i=1}^{n-1} x_{i+1}y_i\right) + x_1y_n\right) - \frac{1}{2}\left(\left(\sum_{i=1}^{n-1} x_1y_{i+1}\right) + x_ny_1\right)$$

Um ein sinnvolles Ergebnis zu erhalten, sollte d_{ABD} noch mit der durchschnittlichen Länge der Trajektorien normalisiert werden.

5.3.14 DISSIM

Wir werden in Abschnitt 6.4 sehen, dass die meisten der bisher vorgestellten Ähnlichkeitsmaße die Zeitkomponente ignorieren – also räumliche statt spatiotemporale Ähnlichkeit messen, wenn sie nicht annehmen, dass die zu vergleichenden Trajektorien die gleiche Länge und die gleichen Zeitintervalle haben.

Frentzos et al. stellen 2007 ein neues Unähnlichkeitsmaß vor, das sie als **DISSIM** (von engl. *dissimilarity*) bezeichnen und das tatsächlich die spatiotemporale Distanz von Trajektorien berechnet [FGT07]. Es definiert sich durch das Integral der euklidischen Distanz in einem validen Zeitintervall:

Definition 5.24 (DISSIM) *Seien σ und τ Trajektorien der Länge n und m. Gelte außerdem für die Zeitstempel $t_{\sigma,1} = t_{\tau,1}$ und $t_{\sigma,n} = t_{\tau,m}$. **DISSIM** ist definiert durch:*

$$d_{DISSIM}(\sigma,\tau) = \sum_{i=1}^{L-1} \int_{t_i}^{t_{i+1}} D(t)\, dt$$

wobei t_i *die Zeitstempel von* σ *oder* τ *sind und* L *die Anzahl der unterschiedlichen diskreten Zeitstempel ist:* $\max(n, m) \leqslant L \leqslant n + m$. *Die euklidische Distanz* D *zwischen zwei Punkten, die sich linear bewegen, lässt sich laut [FGPT07] beschreiben als:*

$$D(t) = \sqrt{at^2 + bt + c}$$

Wenn ein Zeitstempel der einen Trajektorie in der anderen nicht existiert, wird (lineare) Interpolation verwendet. Bei der Berechnung des Integrals wird eine Fallunterscheidung vorgenommen: Für den Fall, dass $a = 0$ (dann ist – wie in [FGPT07] gezeigt – auch $b = 0$) erhalten wir:

$$\int_{t_i}^{t_{i+1}} D(t)\, dt = \frac{\sqrt{c}}{t_{i+1} - t_i}$$

Andernfalls ($a > 0$):

$$\int_{t_i}^{t_{i+1}} D(t)\, dt = \left| \frac{2at + b}{4a} \sqrt{at^2 + bt + c} - \frac{b^2 - 4ac}{8a\sqrt{a}} \left(\frac{2at + b}{\sqrt{4ac - b^2}} \right) \right|_{t_i}^{t_{i+1}}$$

Effektiv wird bei DISSIM, genau wie bei ABD, die Unähnlichkeit der Trajektorien durch Flächenberechnung ermittelt. Die Translation zur Minimierung findet jedoch nicht statt. Es werden auch nicht Teilflächen voneinander getrennt und Zyklen entfernt. Der wesentliche Unterschied besteht nämlich darin, dass nicht wie bei ABD die tatsächliche Fläche zwischen den Trajektorien im Raum berechnet wird – was einem Integral in der Raumdimension entspricht –, sondern nach der Zeit integriert wird.

Die Definition von d_{DISSIM} ist abgeleitet von der Distanz für kontinuierliche Trajektorien, die keine Summe, sondern ausschließlich ein Integral für das gesamte Zeitintervall verwendet [FGT07]. Ganz ähnlich verfahren auch andere Autoren: Auria et al. definieren die Distanz zwischen zweier Trajektorien ebenfalls als Integral einer euklidischen Punktdistanzfunktion über die Zeit, normalisieren diesen Wert allerdings noch mit der Länge des Zeitintervalls [DN05].

5.3.15 Sequence-Pattern-Mining

Viele der bisher vorgestellten Unähnlichkeitsmaße waren in ihrer Komplexität[4] recht überschaubar. Das von Yang et al. in [YCW+12] vorgeschlagene Unähnlichkeitsmaß ist deutlich fortgeschrittener. Der von den Autoren angeführte Anwendungsfall betrifft die in Abschnitt 2.1 vorgestellte Ereigniserkennung in Videos.

[4] Hier nicht im Sinne der mathematischen Komplexität der Berechnung, sondern im klassischen.

Ihre Technik basiert auf der **Segmentierung** von Trajektorien, also deren Zerlegung in ihre Segmente. Die genaue verwendete Technik für die Segmentierung und den Vergleich von Segmenten behandeln wir in Abschnitt 5.5.1. Das verwendete Konzept zur Segmentierung nennt sich **Common-Appearance-Interval** (CAI) und versieht die Segmente mit einer semantischen Information in Abhängigkeit von Ort und Zeit [ZCZC09]. Der Grund für die Segmentierung ist, interessante Ähnlichkeiten von Subtrajektorien erkennen zu können, die nicht erkannt würden, wenn die betreffenden vollständigen Trajektorien sich an anderen Stellen so stark unterscheiden, dass ihre vollständige Ähnlichkeit zu gering ist [YCW+12, Abschn. 2.2].

Das Unähnlichkeitsmaß auf Trajektorien, das Yang et al. vorschlagen, ist eine Summe aus drei voneinander unabhängigen Komponenten. Die erste davon ist die euklidische Distanz d_{AI_RMS}, die anderen werden **Positionsdistanz** (engl. *location distance*) d_l und **Richtungsdistanz** (engl. *direction distance*) d_θ genannt. Während unter Positionsdistanz das Minimum der vertikalen Differenzen zwischen Start- und Endpunkten der Trajektorien verstanden wird, definiert sich die Richtungsdistanz über die Winkel der Strecken. Weil ihr Unähnlichkeitsmaß nur Trajektorien der gleichen Länge akzeptiert, schlagen die Autoren vor, die längere der beiden schlicht auf die Länge der kürzeren zu trimmen, also die überschüssigen Glieder zu ignorieren.

Definition 5.25 (Sequence-Pattern-Mining) *Seien σ und τ Trajektorien der Länge n. Das von Yang et al. vorgeschlagene Unähnlichkeitsmaß im Rahmen von **Sequence-Pattern-Mining** (SPM) ist definiert als:*

$$d_{SPM}(\sigma, \tau) = d_{AI\_RMS}(\sigma, \tau) + \delta d_l + \eta d_\theta$$

wobei

$$d_l = \min\{|\sigma[0]_x - \tau[0]_x|, |\sigma[n]_x - \tau[n]_x|\}$$

$$d_\theta = \frac{1}{n} \sum_{i=1}^{n} \left(1 - \cos\left(\frac{\sigma[i] \cdot \tau[i]}{\|\sigma[i]\| \cdot \|\tau[i]\|}\right)\right)$$

Wir bezeichnen mit $\sigma[i]_x$ die vertikale Koordinate einer Position $\sigma[i]$ und mit $\|\sigma[i]\|$ die euklidische Länge einer Strecke von $\sigma[i-1]$ bis $\sigma[i]$. Das Unähnlichkeitsmaß erlaubt es, mit Gewichtungen δ und η den Einfluss von Positionsdistanz und Richtungsdistanz in das Gesamtergebnis zu bestimmen.

In einem Clustering-Experiment behauptet sich d_{SPM} gegen die euklidische Distanz, gegen LCSS, sowie gegen die Hausdorff-Distanz [YCW+12, Kap. 4].

5.3.16 AAL-Warping

Vlachos et al., die zwei Jahre zuvor LCSS auf Trajektorien angewandt hatten, verfolgen 2004 einen neuen Ansatz. Ihr Ziel ist es, eine Vergleichstechnik für die Anwendung der Handschrifterkennung zu formulieren, die resistent gegen Rotation der Trajektorien ist [VGD04].

Zu diesem Zweck machen sie Gebrauch von einem Merkmalsraum (Abschnitt 4.2.4), der die gewünschte Eigenschaft der Rotationsinvarianz hat. Weil die Merkmale den Winkel und die euklidische Länge von Strecken umfassen, wird der Raum *angle-/arc-length space* (**AAL**-Raum) genannt. In diesem Raum wird dann die Unähnlichkeit mit dem bekannten DTW – also einem Warping-Verfahren – berechnet. Ähnlich wie bei s_{LCSS2} (Abschnitt 5.3.7), soll die Distanz zwischen zwei Trajektorien minimiert werden, indem optimale Transformationen der einen Trajektorie zugelassen werden. Dies können wir allgemein so formulieren:

Definition 5.26 (Unter Transformationen invariante Distanz) *Sei* d *eine Distanzfunktion auf Trajektorien und* $\mathcal{R}$ *die Menge aller Funktionen, die eine oder mehrere der Transformationen Rotation, Skalierung und Translation ausführen. Dann ist die* ***unter*** *all solchen* ***Transformationen invariante Distanz*** *zwischen zwei Trajektorien* σ *und* τ *definiert als:* $d'(\sigma, \tau) = \min_{r \in \mathcal{R}} d(\sigma, r(\tau))$.

Anders ausgedrückt ist eine unter Transformationen invariante Distanz translationsinvariant (Definition 4.8), rotationinvariant (Definition 4.9) und skalierungsinvariant (Definition 4.10). Genau eine solche Distanz soll das Ziel bei Vlachos et al. sein. Sie ist jedoch offensichtlich sehr aufwendig zu berechnen. Das ist der Grund für die Verwendung eines Merkmalsraumes.

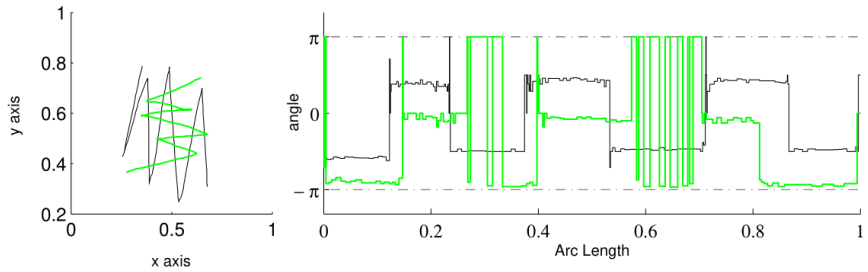


Abbildung 5.8: Zwei Trajektorien im euklidischen und im AAL-Raum [VGD04].

Eine Trajektorie wird transformiert in eine Sequenz von Paaren aus Drehwinkel und Länge für all ihre Strecken. Der Drehwinkel (engl. *turning angle*) eignet sich besonders gut dafür, die von Menschen wahrgenommene Ähnlichkeit von zweidimensionalen Formen abzubilden [SAF94]. Eine Trajektorie wird also zunächst in ihre Strecken zerlegt, wobei

jede Strecke als Vektor betrachtet wird. Der Drehwinkel wird wie in der Mathematik üblich mit Arkuskosinus und Skalarprodukt zwischen dem Vektor und dem Basisvektor der horizontalen Raumdimension berechnet, ganz ähnlich wie in Definition 5.25. Je nach Richtung der Rotation wird noch das Vorzeichen geändert. Zusammen mit der Länge dieser Vektoren bildet eine Folge solcher Paare die Repräsentation einer Trajektorie im AAL-Raum.

Definition 5.27 (AAL-Warping-Distanz) *Seien* σ *und* τ *Trajektorien. Die* ***AAL-Warping-Distanz*** *ist definiert durch:*

$$d_{\mathrm{AAL}}(\sigma,\tau) = \min \begin{cases} d_{\mathrm{DTW}}(\sigma',\tau'), \\ 2\pi - d_{\mathrm{DTW}}(\sigma',\tau') \end{cases}$$

wobei σ' *und* τ' *die AAL-Repräsentationen von* σ *und* τ *mit (ggf. durch Interpolation) gleicher räumlicher Länge der einzelnen Strecken sind.*

5.3.17 Envelope-Technik

Agrawal et al. erkennen schon 1995 das Problem der Lücken bei der Berechnung der Ähnlichkeit zwischen Zeitreihen, das auch von LCSS gelöst wird [LS95]. Sie schlagen ein Ähnlichkeitsmodell vor, das ähnliche Zeitreihen erkennt, auch wenn eine mitten darin liegende Reihe von Gliedern nicht gepaart wird [KGP01, Kap. 1]. Intuitiv sind zwei Zeitreihen hierbei ähnlich, wenn die eine von einer Hülle (engl. *envelope*) einer bestimmten Breite um die andere eingeschlossen werden kann. Dieses Ähnlichkeitsmaß ist nach Angaben der Autoren in der Lage, mit Rauschen, Skalierung und Translation umzugehen.

Definition 5.28 (Envelope-Ähnlichkeit) *Seien* T_1 *und* T_2 *zwei Zeitreihen der Länge* n *und* $\zeta \in [0,1]$ *eine Schranke.* T_1 *und* T_2 *gelten gemäß der* ***Envelope-Ähnlichkeit*** *nach Agrawal et al. als ähnlich, genau dann, wenn sie sich nicht überschneidende Teilsequenzen* S_1 *und* S_2 *enthalten, sodass gilt:*

- $S_1[i] < S_1[j]$ *und* $S_2[i] < S_2[j]$ *für* $1 \leqslant i < j \leqslant n$*,*
- *eine Verschiebung* a *und eine Skalierung* λ *existiert, sodass gilt:*

$$\forall_{i=1}^{n}((S_1[i] * \lambda) \oplus a) \simeq S_2[i]$$

  wobei $\simeq$ *ein Teilsequenz-Ähnlichkeitsoperator (engl.* subsequence similarity operator*) ist, und*
- *die Längen der Teilsequenzen* $|S_*|$ *einen ausreichenden Anteil der Länge der gesamten Zeitreihen* $|T_*|$ *ausmachen:* $\frac{\sum |S_1| + \sum |S_2|}{|T_1| + |T_2|} \geqslant \zeta$*.*

Der Teilsequenz-Ähnlichkeitsoperator $\simeq$ betrachtet dabei zwei Sequenzen als ähnlich, wenn eine von ihnen die andere mit einer Hülle einer gegebenen Breite ϵ umschließt. Die Ähnlichkeit von Teilsequenzen wird letzlich mit der Maximumsnorm (Abschnitt 5.1) berechnet. Im Gegensatz zu LCSS berücksichtigen die Autoren nur eindimensionale Zeitreihen beziehungsweise Trajektorien [LS95, Abschn. 2].

5.4 Außergewöhnlicher Trajektorienvergleich

In diesem Abschnitt stellen wir einige Techniken vor, die sich deutlicher von den bisher vorgestellten unterscheiden. Teilweise verwenden sie interessante Merkmalsräume oder eine speziellere Art von Trajektorien. Um unsere Arbeit nicht auf triviale Ähnlichkeitsmaße zu beschränken und unsere Klassifizierung abstrakt zu gestalten, führen wir solche Techniken hier auf und untersuchen sie ebenfalls.

5.4.1 Spatial-Assembling-Distance

Man kann Zeitreihen vergleichen, indem man zueinander passende Segmente – sogenannte *Patterns* – findet und das Ähnlichkeitsproblem auf das Finden der ähnlichsten Menge von Patterns reduziert. Diese Technik wird als **Spatial-Assembling-Distance** (SpADe) bezeichnet [CNOT07].

SpADe ist durch den Bedarf motiviert, resistent gegen Transformationen in der Raumdimension zu sein, was in der Publikation als *amplitude shifting* bezeichnet wird. Wir werden in Abschnitt 6.11 sehen, was es damit auf sich hat. Der von den Autoren angeführte Anwendungsfall ist die Erkennung von Mustern (engl. *patterns*) in Zeitreihen.

Bei diesem Ansatz werden aus Zeitreihen zunächst lokale Patterns einer fixen Länge w extrahiert, was von einer als *General Match* [MWH02] bekannten Methode inspiriert ist. Solche lokalen Patterns haben die Form $\mathtt{lp} = \langle \theta_{\mathtt{pos}}, \theta_{\mathtt{amp}}, \theta_{\mathtt{shp}}, \theta_{\mathtt{tscl}}, \theta_{\mathtt{ascl}} \rangle$ mit Werten für Position, durchschnittliche Amplitude, Formsignatur und zeitlicher und räumlicher Skalierung. Die Distanz $\mathtt{d}_{\mathtt{lp}}$ für zwei lokale Patterns berechnet sich als gewichtete Summe der Differenzen von $\theta_{\mathtt{amp}}$ und $\theta_{\mathtt{shp}}$. Die Autoren sprechen davon, dass ein *Match* zwischen zwei lokalen Patterns $\mathtt{lp}_1$ und $\mathtt{lp}_2$ genau dann vorliegt, wenn deren Distanz unterhalb eines gegebenen Grenzwertes liegt, also $\mathtt{d}_{\mathtt{lp}}(\mathtt{lp}_1, \mathtt{lp}_2) < \epsilon$. Für zwei gegebene Zeitreihen der Länge n beziehungsweise m werden dann deren Matches von lokalen Pattern in eine sogenannte **Matching-Matrix** eingetragen. Diese Matching-Matrix ist so breit wie die erste Zeitreihe und so hoch wie die zweite, hat also n Spalten und m Zeilen. Ein **Pfad** durch eine solche Matrix ist eine Folge von adjazenten Elementen, die bei

$(0,0)$ beginnt und (n,m) endet. Für einen Pfad kann entsprechend der Inhalte der Matrix eine Länge (die Kosten) berechnet werden. Die Distanz der Zeitreihen ist schließlich anschaulich die Länge des kürzesten Pfades durch deren Matching-Matrix.

Definition 5.29 (Spatial-Assembling-Distance) *Seien* T_1 *und* T_2 *zwei Zeitreihen der Länge* n *und* m. *Seien außerdem* g *eine monoton steigende Funktion, die die Lücken zwischen zwei lokalen Patternmatches bestraft, und* h *eine Funktion, die die Unterschiede zwischen zwei Pattern in Translation und Skalierung gewichtet bestraft.* R *bezeichne alle möglichen Pfade* $\langle r_1, r_2, \dots r_l \rangle$ *durch die mit dem Grenzwert* ϵ *erzeugte Matching-Matrix von* T_1 *und* T_2. *Die Spatial-Assembling-Distance (**SpADe**) ist definiert als:*

$$d_{\mathrm{SpADe}}(T_1, T_2, \epsilon, g, h) = \min_{r \in R} \left(D_{\mathrm{start}}(r_1, r_2) + \sum_{i=2}^{l-2} D(r_i, r_{i+1}) + D_{\mathrm{end}}(r_{l-1}, r_l) \right)$$

wobei

$$D(r_a, r_b) = g\left(ED_x(r_a, r_b)\right) + g\left(ED_y(r_a, r_b)\right) + h\left(r_a, r_b\right)$$

$$D_{\mathrm{start}}(r_1, r_2) = g\left(r_2[x] - \frac{w}{2}\right) + g\left(r_2[y] - r_1[y] - \frac{w}{2}\right)$$

$$D_{\mathrm{end}}(r_{l-1}, r_l) = g\left(m - r_{l-1}[x] - \frac{w}{2}\right) + g\left(r_l[y] - r_{l-1}[y] - \frac{w}{2}\right)$$

$$ED_x(r_a, r_b) = \begin{cases} \max(p_a[x] - r_b[x] - w, 0) & , r_a[x] > r_b[x] \\ \infty & , \textit{sonst} \end{cases}$$

$$ED_y(r_a, r_b) = \begin{cases} \max(p_a[y] - r_b[y] - w, 0) & , r_a[y] > r_b[y] \\ \infty & , \textit{sonst} \end{cases}$$

5.4.2 Similarity-search-based-on-Threshold-Queries

Similarity-search-based-on-Threshold-Queries (TQuEST) ist ein neuartiger Ansatz für den Vergleich von Zeitreihen [AKK+06]. Die Technik beruht auf der Einsicht, dass es für die Erkennung von zeitlichen Abhängigkeiten oft ausreichend ist, die Zeitpunkte zu kennen, an denen die Werte der Zeitreihen eine bestimmte Schwelle (engl. *threshold*) überschreiten (Abbildung 5.9). Eine Anfrage nach ähnlichen Zeitreihen zur eingegebenen Zeitreihe T wird um eine Schwelle θ erweitert (engl. *threshold query*). Nun wird die Anfrage beantwortet, indem sowohl T als auch die Zeitreihen der Datenbank in Zeitintervalle zerlegt werden, in denen die Werte größer als θ sind (engl. *threshold-crossing time intervals*, TCT). Jedes Zeitintervall wird dann als zweidimensionaler Punkt betrachtet, wobei Start- und Endzeitstempel die beiden Dimensionen bilden. Es hat die

Form $\langle t_1, t_2 \rangle$, wobei t_1 und t_2 Zeitstempel sind. Das Ähnlichkeitsmaß für zwei Zeitreihen ist schließlich als die Minkowski-Summe dieser Zeitintervalle definiert und ist Grundlage für die TQuEST-Distanz [WMD+13].

Definition 5.30 (TQuEST-Distanz) *Seien* T_1 *und* T_2 *zwei Zeitreihen und* S_1 *und* S_2 *die jeweils dazugehörigen TCT für eine bestimmte Schwelle* θ*. Dann ist die* ***TQuEST-Distanz*** *(auch* Threshold-Distanz*) definiert als:*

$$d_{\text{TQuEST}}(T_1, T_2) = \frac{1}{|S_1|} \sum_{s_1 \in S_1} \min_{s_2 \in S_2} d_{\text{TCT}}(s_1, s_2) + \frac{1}{|S_2|} \sum_{s_2 \in S_2} \min_{s_1 \in S_1} d_{\text{TCT}}(s_2, s_1)$$

wobei die Distanz zwischen zwei Zeitintervallen (TCT) sich wie folgt ergibt:

$$d_{\text{TCT}}(\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle) = \sqrt{(a_1 - a_2)^2 + (b_1 - b_2)^2}$$

Die Definition lässt eine Inspiration durch die Hausdorff-Distanz erahnen. Es wird für jedes TCT der ersten Zeitreihe die minimale Distanz zu einem TCT der zweiten Zeitreihe berechnet. Von diesen Ergebnissen wird jedoch nicht das Supremum ermittelt, sondern das arithmetische Mittel. Die Threshold-Distanz ergibt sich aus der Summe dieser Mittelwerte für beide Zeitreihen. Eine Anfrage mit Schwellenwert $\langle T, \theta, k \rangle$ liefert ein Ergebnis X mit mindestens k Zeitreihen aus der Datenbank, für die gilt, dass deren TQuEST-Distanz zur Zeitreihe der Anfrage jeweils kleiner ist als die TQuEST-Distanz zu jeder anderen Zeitreihe, die nicht im Ergebnis ist: $\forall x \in X \; \forall y \in \overline{X}$: $d_{\text{TQuEST}}(x, T) < d_{\text{TQuEST}}(y, T)$.

Diese Technik entfaltet ihre Stärke, wenn es um die Laufzeit geht, weil für die Beantwortung der Anfrage nicht auf die gesamte Zeitreihe zugegriffen werden muss. Stattdessen reicht die Information über die Zeitpunkte, an denen die Zeitreihen den Schwellenwert überschreiten [AKK+06, Kap. 4]. Wie in Abschnitt 4.2.3 erwähnt, können Zeitreihen – zum Beispiel mit GEMINI – in ihrer Dimension reduziert werden und als Objekte eines Merkmalsraumes gespeichert werden. Im Falle von TQuEST ist dies nicht anwendbar,

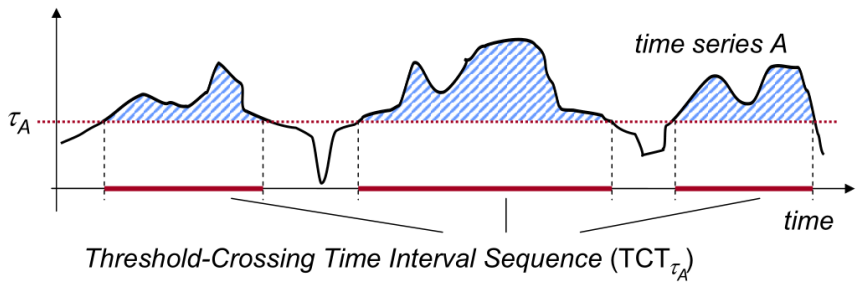


Abbildung 5.9: Threshold-Crossing-Time-Intervals [AKK+06].

da die Zeitkomponenten verloren gehen. Die TCT lassen sich dann nicht mehr berechnen [AKK+06, Kap. 2]. Die Autoren sehen in ihrer Publikation nur eindimensionale Zeitreihen beziehungsweise Trajektorien vor. Da sich ihre Technik nicht ohne Weiteres auf mehrdimensionale Trajektorien anwenden lässt, ist sie für uns nur eingeschränkt interessant.

5.4.3 Netzwerk-basierter Ansatz

Hwang et al. bemerken in [HKL05, HKL06], dass in vielen Anwendungsfällen die bewegten Objekte, deren Trajektorien wir betrachten, sich nicht frei im Raum bewegen können. Vielmehr seien ihre möglichen Bewegungen auf zuvor definierte Pfade beschränkt, die man als eine Art Straßennetz (engl. *road network*, RN) modellieren kann. Demzufolge sei es auch nicht zielführend, in solchen Fällen etwa die euklidische Distanz oder die Hausdorff-Distanz zu verwenden, weil diese implizit freie Bewegungsmöglichkeiten annehmen. Die Autoren schlagen daher eine neue Technik vor, um Trajektorien zu vergleichen, die wir als **RN-Technik** bezeichnen. Trajektorien werden in diesem Kontext als Sequenz von Tripeln der Form $\langle \mathsf{SegID}, \mathtt{offset}, \mathsf{t} \rangle$ definiert, wobei SegID ein Identifikator eines Straßensektors, $\mathtt{offset}$ der Versatz zu dessen Startpunkt und t ein Zeitstempel ist [HKL06, Abschn. 2.2]. Weiter gehen sie von vordefinierten Orten – **Points-of-Interest** (PoI) – oder Zeitpunkten – **Times-of-Interest** (ToI) – aus, die für die Anwendung relevant sind. Der Ansatz, für eine Query-Trajektorie ähnliche Trajektorien zu finden, besteht nun aus zwei Schritten, um die Performance zu optimieren: erstens einer Filterung, die Kandidaten für ähnliche Trajektorien aussortiert, und zweitens einer Verfeinerung (engl. *refining*), die für die übrig gebliebenen Kandidaten jeweils die genaue Distanz berechnet. Bei gegebenen PoI findet die Filterung räumlich und die Distanzberechnung zeitlich statt. Bei gegebenen ToI findet die Filterung zeitlich und die Distanzberechnung räumlich statt. Zusätzlich gibt es die Möglichkeit, sowohl Filterung als auch Distanzberechnung spatiotemporal, also räumlich und zeitlich, durchzuführen [HKL06, Abschn. 3]. Die Filterung geschieht mithilfe der Ähnlichkeitsfunktionen, die wie folgt definiert sind:

Definition 5.31 (Road-Network-Ähnlichkeit) *Seien σ und τ zwei wie zuvor definierte Trajektorien. Außerdem sei P eine Menge von PoI und T eine Menge von ToI. Dann ist die* ***räumliche Road-Network-Ähnlichkeit*** *(RN-Ähnlichkeit) definiert als:*

$$s_{\mathrm{RN\_PoI}}(\sigma, \tau, \mathsf{P}) = \begin{cases} 1 & , \forall\, \mathsf{p} \in \mathsf{P}\colon \mathsf{p} \textit{ ist in } \sigma \textit{ und } \tau \\ 0 & , \textit{sonst} \end{cases}$$

*Die **zeitliche Road-Network-Ähnlichkeit** ist definiert als:*

$$s_{RN\_ToI}(\sigma, \tau, T) = \begin{cases} 1 & , \forall t \in T\colon t \in \sigma_T \wedge t \in \tau_T \\ 0 & , \textit{sonst} \end{cases}$$

wobei σ_T und τ_T die Zeitintervalle der Trajektorien von ihrem frühesten bis zu ihrem spätesten Zeitpunkt sind.

Beide Trajektorien müssen also alle PoI beziehungsweise ToI enthalten, damit sie als ähnlich gelten und nicht ausgefiltert werden. Die anschließende Distanzberechnung folgt dieser Definition:

Definition 5.32 (Road-Network-Distanz) *Seien σ, τ, P, T wie oben gegeben. Dann ist die **zeitliche Road-Network-Distanz** (RN-Distanz) definiert als:*

$$d_{RN\_T}(\sigma, \tau, P) = L_p(\sigma, \tau) = \Big(\sum_{p_i \in P} |p_i(\sigma) - p_i(\tau)|^p \Big)^{\frac{1}{p}}$$

*wobei $|p_i(\sigma) - p_i(\tau)|$ die Länge des Zeitraumes zwischen dem Passieren von σ durch p_i und τ durch p_i ist. Passieren die Trajektorien p_i nicht, wird der Wert als unendlich angenommen. Sei nun d eine Distanz auf Positionen im RN. Die **räumliche Road-Network-Distanz** ist definiert als:*

$$d_{RN\_S}(\sigma, \tau, T) = \sum_{t_i \in T} d(\sigma(t_i), \tau(t_i))$$

*Die **spatiotemporale Road-Network-Distanz** ist definiert als:*

$$d_{RN\_TS}(\sigma, \tau, P, T) = d_{RN\_T}(\sigma, \tau, P) + d_{RN\_S}(\sigma, \tau, T)$$

Letztere ist benutzbar, wenn eine Äquivalenz von Zeit und Raum der Form $1\ s = \zeta\ m$ durch einen Parameter ζ definiert wird.

5.4.4 Graph-basierter Ansatz

Tiakas et al. kritisieren in [TPN+09] eine Vielzahl von Eigenschaften der RN-Technik. Zum Beispiel setzt sie voraus, dass ähnliche Trajektorien gemeinsame Positionen haben. Die Autoren definieren eine andere Technik, die die aufgezeigten Schwächen nicht hat. Dabei sind Positionen von Trajektorien als Knoten eines gewöhnlichen Graphen definiert.

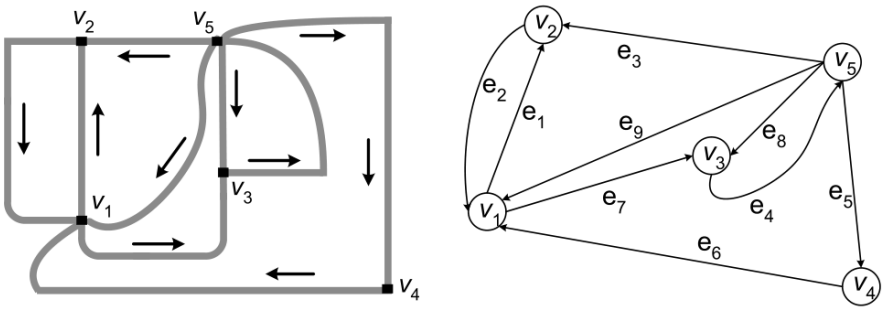


Abbildung 5.10: Road-Network- und Graph-Repräsentation von Trajektorien [TPN+09].

Definition 5.33 (Graph-basierte Distanz) *Seien* σ *und* τ *zwei Trajektorien der Länge* n*, deren Positionen Knoten eines Graphen sind. Dann definieren wir deren* ***Graph-basierte Distanz*** *als:*

$$d_{\text{Graph}}(\sigma,\tau) = \frac{1}{n}\sum_{i=1}^{n} \begin{cases} 0 & , \sigma[i] = \tau[i] \\ \frac{c(\sigma[i],\tau[i])+c(\tau[i],\sigma[i])}{2D_G} & , \textit{sonst} \end{cases}$$

wobei $c(u,v)$ *die Kostenfunktion im Graphen, um von Knoten* u *zu Knoten* v *zu gelangen, und* D_G *der Durchmesser des Graphen ist.*

Die Beschränkung, dass die Trajektorien die gleiche Länge haben müssen, wird aufgehoben, indem sie in Subtrajektorien zerlegt werden. Eine Trajektorie ist genau dann ähnlich zu einer anderen, wenn mindestens eine ihrer Subtrajektorien ähnlich zu mindestens einer Subtrajektorie der anderen ist [TPN+09, Abschn. 4.3].

5.4.5 Grid-basierter Ansatz

Meratnia und de By forschen auf dem Gebiet der Datenkompression für Trajektorien [MdB02, MR04]. Sie schlagen die Aggregation von Trajektorien vor, um viele ähnliche Trajektorien zu einer zusammenzufassen. Dafür eignet sich die Verwendung einer Trajektoriendistanz wie d_{CPD} in Verbindung mit einem Schwellwert aus zwei Gründen nicht: Zum einen ist sie nicht transitiv, das heißt aus geringer Distanz von τ_1 zu τ_2 und von τ_2 zu τ_3 kann man nicht auf geringe Distanz von τ_1 zu τ_3 schließen. Zum anderen ist die Berechnung von d_{CPD} mit einer Zeitkomplexität von $\mathcal{O}(n^2)$ recht teuer [MdB02, Abschn. 2.1].[5] Daher verwenden sie ein **Grid**, also ein Raster von homogenen räumlichen Flächen, das auf den Raum der Trajektorien gelegt wird. Eine Zelle des Grids ist mit einer Ganzzahl versehen, die verrät, wie oft Trajektorien sie durchqueren. Diese Idee minimiert

[5] Auf diesen Fakt werden wir in Abschnitt 6.8 genauer eingehen.

zwar die Menge der Daten, bringt jedoch zahlreiche Probleme mit sich, so etwa die schwierige Wahl der Größe der Zellen [TPN+09, Abschn. 2] oder die mangelnde Unterstützung für die Suche nach ähnlichen Trajektorien [LS05, Abschn. 2].

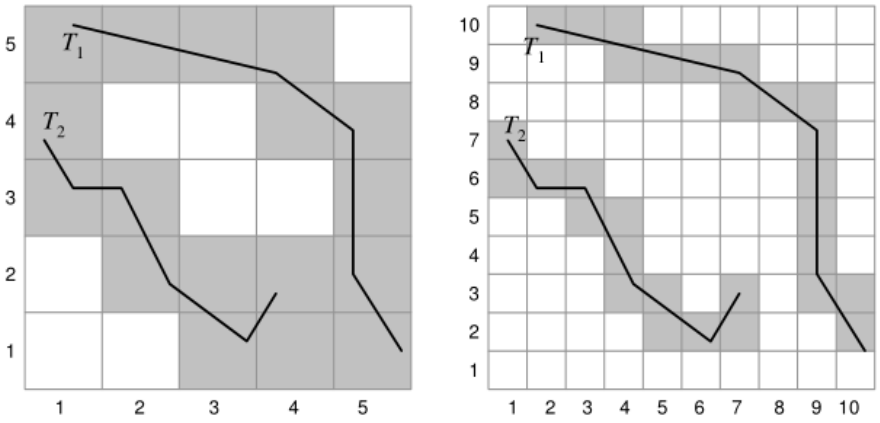


Abbildung 5.11: Zwei Trajektorien im Grid [LS05].

Lin und Su definieren Ähnlichkeitsmaße sowohl auf regulären Trajektorien, als auch auf Trajektorien in Grid-Repräsentation. Das reguläre Ähnlichkeitsmaß entspricht dabei genau d_{AP_mean} beziehungsweise der One-Way-Distance (Definition 5.8), das Ähnlichkeitsmaß im Grid orientiert sich an ihr [LS05]. Eine **Grid-Trajektorie** ist eine Sequenz von Grid-Zellen $\langle g_1, \dots g_n \rangle$.

Definition 5.34 (Grid-basierte Distanz) *Seien* σ *und* τ *zwei Grid-Trajektorien der Länge* n *beziehungsweise* m*. Die* ***Grid-basierte Distanz*** *ist definiert als:*

$$d_{Grid}(\sigma, \tau) = \frac{1}{2} \left(d'_{Grid}(\sigma, \tau) + d'_{Grid}(\tau, \sigma) \right)$$

wobei

$$d'_{Grid}(\sigma, \tau) = \frac{1}{|\sigma|} \sum_{i=1}^{|\sigma|} d_{PTD\_Grid}(\sigma[i], \tau)$$

Dabei bezeichnet $|\sigma|$ *die Länge von* σ*, also* n *beziehungsweise* m*, und* $d_{PTD\_Grid}(\sigma[i], \tau)$ *analog zu* d_{PTD} *(Definition 5.3) die kürzeste euklidische Distanz (Abschn. 5.1) zwischen der Grid-Zelle* $\sigma[i]$ *und allen Grid-Zellen von* τ*.*

Laut Frentzos et al. ist diese Technik DTW überlegen [FGT07, Kap. 2].

5.4.6 Point-Distribution-Model

Ein sehr außergewöhnlicher Ansatz verwendet **Point-Distribution-Models** (PDM) für den Vergleich von Trajektorien. Roduit et al. nutzen eine Menge von Punkten, um Trajektorien mobiler Roboter zu beschreiben und miteinander zu vergleichen. Diese Punkte werden ermittelt, indem Samples von den Positionen der Roboter bei der Durchfahrt durch Tore in einem Parcours erzeugt werden. Sie stellen für diesen Anwendungsfall einen deutlichen Vorteil dieses Verfahrens gegenüber euklidischen Distanzfunktionen fest [RMJ07].

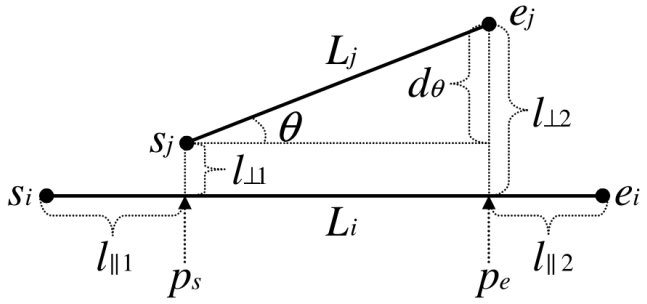
5.4.7 Hidden-Markov-Model

Ebenfalls außergewöhnlich ist die Nutzung eines **Hidden-Markov-Model** (HMM) zur Berechnung der Ähnlichkeit von Trajektorien. Porikli leitet von diesem stochastischen Modell eine Reihe von Unähnlichkeitsmaßen ab, die die Unähnlichkeit von Trajektorien messen, indem sie die Übereinstimmungen ihrer Entsprechungen im HMM-Modell quantifizieren. Er postuliert die Überlegenheit dieser Methode, weil sie Ähnlichkeit von Koordinaten, Ausrichtung und Geschwindigkeit besser als „konventionelle“ Methoden erkennt [Por04]. Mit einem solchen Ansatz ist er nicht alleine [OTC09].

5.5 Sonstige Vergleiche

5.5.1 Segmentvergleich

Wie in Abschnitt 5.3.15 erläutert, kann es sinnvoll sein, Trajektorien zu segmentieren. Die Beobachtung von Lee et al. lautet sinngemäß, dass bei der Clusteranalyse gesamter Trajektorien keine Ähnlichkeiten von Subtrajektorien festgestellt werden können. Lange Trajektorien könnten zueinander sehr ähnliche Subtrajektorien besitzen, obwohl sie in ihrer Gesamtheit sehr unähnlich sind. Ihre Schlussfolgerung ist der Entwurf eines Algorithmus namens **TRACULUS**, der Trajektorien segmentiert, bevor er sie clustert [LHW07]. Kern des Algorithmus ist ein Unähnlichkeitsmaß auf den kürzesten Segmenten, also den Strecken von Trajektorien. Dieses Unähnlichkeitsmaß besteht aus den drei Komponenten **senkrechter Abstand** (engl. *perpendicular distance*), **paralleler Abstand** (engl. *parallel distance*) und **Winkelabstand** (engl. *angle distance*). Ihre Definition ist anschaulich der Abbildung 5.12 zu entnehmen. Der Abstand zwischen zwei Strecken von Trajektorien ist definiert durch die gewichtete Summe dieser drei Abstände.



$$d_\perp = \frac{l_{\perp 1}{}^2 + l_{\perp 2}{}^2}{l_{\perp 1} + l_{\perp 2}}$$

$$d_\parallel = \mathrm{MIN}(l_{\parallel 1}, l_{\parallel 2})$$

$$d_\theta = \|L_j\| \times \sin(\theta)$$

Abbildung 5.12: Vergleich von Segmenten/Strecken von Trajektorien [LHW07].

5.5.2 Benutzervergleich

Maximal-Semantic-Trajectory-Pattern

Ying et al. verfolgen einen Ansatz mit höherem Abstraktionsgrad. Sie schlagen einen Ansatz vor, um die Ähnlichkeit von Benutzern in LBSN zu ermitteln. Das geschieht anhand einer von ihnen neu entwickelten Technik namens *Maximal Semantic Trajectory Pattern Similarity* (MSTP-Ähnlichkeit, deutsch etwa „maximale Semantische-Trajektorien-Muster-Ähnlichkeit“), die Trajektorien anhand ihrer semantischen Ähnlichkeit vergleicht [YLL+10].

Die Motivation für MSTP liegt in der Erkenntnis, dass die meisten anderen Ansätze geographischer Natur sind und geographische Ähnlichkeit der Trajektorien von Nutzern kein gutes Maß für die Ähnlichkeit ebendieser ist. Daher verwenden sie das von Alvares et al. vorgeschlagene Konzept von *semantischen Trajektorien* [BKA09]. Eine **semantische Trajektorie** besteht aus einer Folge von Orten mit semantischen Tags. Beispiele für solche Tags sind etwa *Schule* oder *Krankenhaus*. Alvares et al. stellen auch eine Methode zur Verfügung, geographische Trajektorien in semantische Trajektorien zu transformieren. In Abbildung 5.13 sehen wir, dass Trajektorie 1 mit Trajektorie 3 aufgrund ihrer Tags eine höhere Ähnlichkeit aufweist als mit Trajektorie 2, obwohl diese ihr geographisch ähnlicher ist als jene.

Es werden zunächst sogenannte **Maximal-Semantic-Trajectory-Patterns** (MSTP) aus den geographischen Trajektorien und geographischen Zellen mit Landmarken generiert, die zum Beispiel der Form $\langle\{\text{unbekannt}\}, \{\text{Schule}, \text{Park}\}, \{\text{Park}\}, \{\text{Krankenhaus}\}\rangle$ entsprechen. Es sind nur *maximale* Patterns von Interesse; das sind solche, die nicht Teil von anderen Patterns sind. In unserem Beispiel wäre $\langle\{\text{Schule}, \text{Park}\}, \{\text{Park}\}\rangle$ zwar ein Pattern, aber kein maximales. Auf solchen Patterns wird dann ein Ähnlichkeitsmaß de-

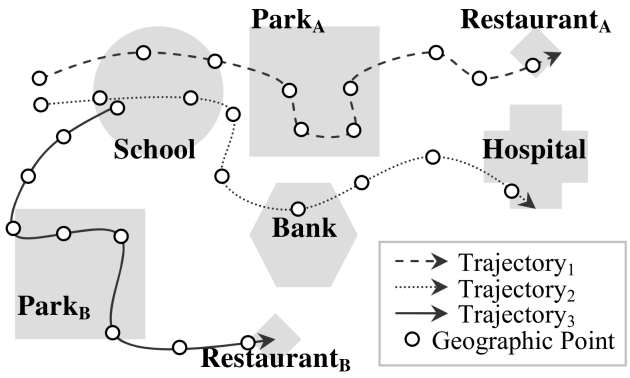


Abbildung 5.13: Ein Beispiel für semantische Trajektorien [YLL+10].

finiert, das wiederum Grundlage für ein darauf definiertes Ähnlichkeitsmaß für Benutzer sein wird. Weil zwei Patterns augenscheinlich umso ähnlicher sind, je mehr gemeinsame Tags sie haben, bedient sich das Ähnlichkeitsmaß dem bekannten LCSS. Beispielsweise ist für die Patterns $P = \langle\{\text{unbekannt}\}, \{\text{Schule}, \text{Park}\}, \{\text{Park}\}, \{\text{Krankenhaus}\}\rangle$ und $Q = \langle\{\text{Schule}\}, \{\text{Bank}\}, \{\text{Schule}\}, \{\text{Krankenhaus}, \text{Bank}\}\rangle$ die längste gemeinsame Teilfolge (engl. *longest common subsequence*, **LCS**) $LCS(P, Q) = \langle\{\text{Schule}\}, \{\text{Krankenhaus}\}\rangle$.

Die Verwendung von Patterns hat gegenüber Lévy-Walk [RSH+11] oder Markow-Ketten den Vorteil, dass sie für die Bestimmung der Ähnlichkeit von Benutzern präziser ist, weil nur für den Nutzer relevante Orte zum Tragen kommen [CLMP14].

Definition 5.35 (MSTP-Anteilsverhältnis) *Seien* P *und* Q *zwei MSTP. Dann ist das* ***Anteilsverhältnis*** *(engl.* participation ratio*) des gemeinsamen Teils mit dem Pattern* P *wie folgt definiert:*

$$\mathrm{ratio}(LCS(P, Q), P) = \frac{\sum_{i=1}^{|P|} \sum_{j=1}^{|LCS(P,Q)|} M(P_i, LCS_j)}{|P|}$$

wobei

$$M(P_i, LCS_j) = \begin{cases} \frac{|P_i \cap LCS_j|}{|P_i|} & , P_i \textit{ ist in } LCS_j \\ 0 & , \textit{ sonst} \end{cases}$$

Nun können wir die Ähnlichkeit zweier MSTP definieren. Entweder wird dafür unmittelbar das durchschnittliche Anteilsverhältnis *Equal Average* (EA) der Patterns berechnet, oder dieses wird in Relation zur Länge der Patterns gesetzt, was mit *Weighted Average* (WA) bezeichnet wird. Letzteres ergibt Sinn, weil längere Patterns eine höhere Ähnlichkeit nahelegen und deswegen eine höhere Gewichtung erfahren.

Definition 5.36 (MSTP-Ähnlichkeit) *Seien* P *und* Q *zwei MSTP. Dann ist deren* ***MSTP-Ähnlichkeit*** *wie folgt definiert:*

$$s_{\text{MSTP\_EA}}(P, Q) = \frac{\text{ratio}(\text{LCS}(P, Q), P) + \text{ratio}(\text{LCS}(P, Q), Q)}{2}$$

sowie

$$s_{\text{MSTP\_WA}}(P, Q) = \frac{|P| \times \text{ratio}(\text{LCS}(P, Q), P) + |Q| \times \text{ratio}(\text{LCS}(P, Q), Q)}{|P| + |Q|}$$

Für zwei Benutzer, die als Menge von MSTP repräsentiert werden, wird schließlich deren Ähnlichkeit definiert als die gewichtete Summe der MSTP-Ähnlichkeit aller möglichen MSTP-Paarungen.

Chen et al. haben den Fehler in [YLL+10] gefunden, dass selbst zwei identische Benutzer nicht den maximalen Ähnlichkeitswert von 1.0 erreichen können, und ihn behoben [CPX13]. Sie schlagen außerdem noch zwei weitere Verbesserungen vor, die zum einen unterschiedliche Wahrscheinlichkeiten von mehreren Tags innerhalb einer Zelle, und zum anderen über den Ort hinausgehende Semantiken betreffen [CPX14]. Diese neue Metrik wird als *Maximal Trajectory Pattern Similarity* (**MTP-Ähnlichkeit**) bezeichnet, während die Originalmetrik *Maximal Semantic Trajectory Pattern Similarity* (MSTP-Ähnlichkeit) heißt [CLMP14].

Mobility Similarity und Location-Semantic Similarity

Chen et al. bemängeln in [CLMP14], dass es bis zum Erscheinen ihrer Publikation keine formalen Prinzipien zur Evaluation eines Ähnlichkeitsmaßes auf Benutzern anhand von Mobilitätsprofilen mit Trajektorien gebe. Sie stellen daher solche Prinzipien auf und schlagen ein Unähnlichkeitsmaß vor, das diese erfüllt. Diese Prinzipien umfassen zum Beispiel dessen Wertebereich als Einheitsintervall, seine Symmetrie, seinen Wert von 1 beim Vergleich eines Mobilitätsprofiles mit sich selbst, sowie Aussagen über sein Verhalten mit einem dritten Mobilitätsprofil. Chen et al. zeigen detailliert, dass MSTP und MTP jene nicht erfüllen [CLMP14, Kap. 3].

Grundlage ihres neuen Unähnlichkeitsmaßes ist wie zuvor das von Giannotti et al. vorgeschlagene Konzept der *Trajectory Patterns* [GNPP07]. Genauer gesagt ist ein **Trajectory Pattern** eine Sequenz von *Regions of Interest*, wobei eine **Region-of-Interest** (RoI) nichts anderes ist als eine Teilmenge der möglichen geographischen Positionen. Es wird ein **Mobilitätsprofil** für Benutzer definiert, das alle Patterns enthält, deren Anteil an all seinen Patterns eine bestimmte untere Schranke überschreitet. Die Benutzerähn-

lichkeit wird dann auf die Ähnlichkeit ihrer Mobilitätsprofile reduziert. Die schließlich vorgeschlagene Metrik berücksichtigt nicht nur die längste gemeinsame Teilfolge (LCS) von Patterns, sondern alle gemeinsamen Patterns der Benutzer. Sie basiert auf dem *Bray-Curtis-Unähnlichkeitsmaß* [BC57], das auf Vektoren mit Werten der Mobilitätsprofile angewendet wird. Ihre neue Metrik bezeichnen die Autoren selbst mit **Common-Pattern-Sets** (CPS). Sie erfüllt die vorgeschlagenen Prinzipien [CLMP14, Abschn. 4.1]. Darüber hinaus wenden Chen et al. ihre Technik nicht nur auf geographische, sondern auch auf semantische Trajektorien – analog zu MSTP – an.

6 Klassifizierung von Vergleichstechniken

Kapitel 5 hat deutlich gezeigt, dass es viele teilweise sehr unterschiedliche Ansätze für den Vergleich von Trajektorien gibt. Um die Auswahl eines geeigneten Ähnlichkeitsmaßes für einen Anwendungszweck zu erleichtern, ist ein objektiver Vergleich zwischen ihnen notwendig.

> „Given the multitude of competitive techniques, we believe that there is a strong need for a comprehensive comparison which [...] may also reveal certain omissions in the comparative observations reported in the individual works. In the common case, every newly-introduced representation method or distance measure has claimed a particular superiority over some of the existing results. However, it has been demonstrated that some empirical evaluations have been inadequate and, worse yet, some of the claims are even contradictory.“ [WMD+13, Kap. 1]

Tatsächlich fehlt es – wie in Kapitel 3 dargelegt – an Maßstäben, nach denen sich Ähnlichkeitsmaße vergleichen lassen. In diesem Kapitel werden wir solche Maßstäbe in Form von Klassifikationen definieren. Sie sind inspiriert von Charakteristika, die sich bei den Ähnlichkeitsmaßen beobachten lassen. Diese Klassifikationen erlauben es uns, sowohl die in Kapitel 5 vorgestellten Ähnlichkeitsmaße als auch alle weiteren, insbesondere noch nicht existierende, miteinander zu vergleichen.

Wir haben in Kapitel 5 jede Technik durch ein explizites Ähnlichkeitsmaß mathematisch und in der Notation konsistent definiert. Dies macht es nun einfacher, die Techniken zu beurteilen, indem wir für die jeweiligen Ähnlichkeitsmaße eine Klassifizierung vornehmen. Konkret umfasst die Menge, die wir genauer untersuchen, die folgenden Ähnlichkeitsmaße: $\mathrm{d}_{\mathrm{CPD}}$ (Definition 5.5), $\mathrm{d}_{\mathrm{AI\_min}}$ (Definition 5.6), $\mathrm{d}_{\mathrm{AI\_max}}$ (Definition 5.6), $\mathrm{d}_{\mathrm{AI\_sum}}$ (Definition 5.6), $\mathrm{d}_{\mathrm{AI\_mean}}$ (Definition 5.6), $\mathrm{d}_{\mathrm{AI\_RMS}}$ (Definition 5.6), $\mathrm{d}_{\mathrm{AI\_median}}$ (Definition 5.6), d_{E} (Definition 5.7), $\mathrm{d}_{\mathrm{AP\_mean}}$ (Definition 5.8), $\mathrm{d}_{\mathrm{Hausdorff}}$ (Definition 5.9), $\mathrm{d}_{\mathrm{MOHD}}$ (Definition 5.10), $\mathrm{d}_{\mathrm{Fréchet}}$ (Definition 5.11), $\mathrm{d}_{\mathrm{DTW}}$ (Definition 5.12), $\mathrm{d}_{\mathrm{LCSS\_1}}$ (Definition 5.15), $\mathrm{d}_{\mathrm{LCSS\_2}}$ (Definition 5.15), $\mathrm{d}_{\mathrm{LCSS\_SM\_1}}$ (Abschnitt 5.3.7), $\mathrm{d}_{\mathrm{LCSS\_SM\_2}}$ (Abschnitt 5.3.7), $\mathrm{d}_{\mathrm{EDR}}$ (Definition 5.17), $\mathrm{d}_{\mathrm{ERP}}$ (Definition 5.18), $\mathrm{s}_{\mathrm{Swale}}$ (Definition 5.19), d_{PF} (Definition 5.20), $\mathrm{d}_{\mathrm{SBD\_TS}}$ (Definition 5.22), $\mathrm{d}_{\mathrm{SBD\_S}}$ (Definition 5.22), $\mathrm{d}_{\mathrm{ABD}}$ (Definition 5.23),

$\mathsf{d}_{\text{DISSIM}}$ (Definition 5.24), d_{SPM} (Definition 5.25), d_{AAL} (Definition 5.27), $\mathsf{d}_{\text{SpADe}}$ (Definition 5.29), $\mathsf{d}_{\text{TQuEST}}$ (Definition 5.30), $\mathsf{d}_{\text{RN\_T}}$ (Definition 5.32), $\mathsf{d}_{\text{RN\_S}}$ (Definition 5.32), $\mathsf{d}_{\text{RN\_TS}}$ (Definition 5.32), $\mathsf{d}_{\text{Graph}}$ (Definition 5.33) und d_{Grid} (Definition 5.34). Eine Übersicht über die Ergebnisse dieser Klassifizierung wird in Abschnitt 7.2 zu finden sein.

Wir schlagen die folgenden Klassifikationen für den Vergleich von Ähnlichkeitsmaßen auf Trajektorien vor:

- **MR**: Verwendung eines Merkmalsraumes (Abschnitt 6.1)
- **ZS**: Anforderungen an Zeitstempel (Abschnitt 6.2)
- **ZI**: Anforderungen an Zeitintervalle (Abschnitt 6.2)
- **LN**: Anforderungen an Länge (Abschnitt 6.2)
- **AK**: Akkumulation der Glieder (Abschnitt 6.3)
- **EL**: Elastizität (Abschnitt 6.3)
- **LE**: Längenempfindlichkeit (Abschnitt 6.4)
- **ZE**: Zeitempfindlichkeit (Abschnitt 6.4)
- **MD**: Maßdimension (Abschnitt 6.5)
- **PRM**: Parametrierbarkeit (Abschnitt 6.6)
- **MTR**: Metrische Eigenschaften (Abschnitt 6.7)
- **KMP**: Zeitkomplexität der Berechnung (Abschnitt 6.8)
- **SI**: Sampling-Invarianz (Abschnitt 6.9)
- **AR**: Empfindlichkeit auf Ausreißer (Abschnitt 6.10)
- **TI**: Transformationsinvarianz (Abschnitt 6.11)
- **IN**: Eignung für inkrementelle Berechnung (Abschnitt 6.12)
- **PRE**: Notwendigkeit der Vorverarbeitung (Abschnitt 6.13)
- **SUB**: Eignung für Subtrajektorien (Abschnitt 6.14)

Innerhalb dieses Kapitels werden wir jeder Klassifikation oder jeder Gruppe von Klassifikationen einen Abschnitt widmen, der jeweils zweigeteilt ist. Im ersten Teil werden jeweils die Klassifikationen selbst beschrieben. Wir legen fest, welche Unterscheidungen wir treffen, also wie Klassengrenzen gezogen werden. Wir sprechen von der *Klassifizierung*. In dem zweiten Teil ordnen wir die in Kapitel 5 vorgestellten Techniken beziehungsweise Ähnlichkeitsmaße in die festgelegten Klassen ein. Wir sprechen von der *Klassierung*.

Angenommen, es gibt für eine Klassifikation X die Merkmale A, B und C, dann bezeichnen wir die dazugehörigen Klassen von Ähnlichkeitsmaßen mit X_A, X_B und X_C. Wenn ein Ähnlichkeitsmaß d die Kriterien für A erfüllt, ist es Element der entsprechenden Klasse: $\mathsf{d} \in \mathsf{X}_\mathsf{A}$. Erfüllt es sie nicht, ist es in der Komplementärmenge: $\mathsf{d} \in \overline{\mathsf{X}_\mathsf{A}}$. Die definierten Klassen müssen nicht zwangsweise disjunkt sein, auch wenn sie das in den meisten Fällen sind.

6.1 Raum

Klassifizierung

Das wohl grundlegendste Charakteristikum eines Ähnlichkeitsmaßes ist der Raum, in dem es operiert. Davon hängt auch ab, ob es überhaupt sinnvoll ist, andere Charakteristika miteinander zu vergleichen. Wir können prinzipiell jedes Ähnlichkeitsmaß, das einen bestimmten Raum nutzt, einer neuen Klasse für alle Ähnlichkeitsmaße in diesem Raum zuweisen. Darüber hinaus ist es aber interessant, sich für ein Ähnlichkeitsmaß die folgenden Dinge klarzumachen:

- Erstens, ob der Wert für die Ähnlichkeit direkt auf den Positionen und Zeitstempeln der Trajektorien – den Primärdaten (engl. *raw data*) – berechnet wird, oder in einem Merkmalsraum (Abschnitt 4.2.4). Ist letzteres der Fall, ordnen wir das Ähnlichkeitsmaß der Klasse MR_{M} zu, andernfalls der Komplementärmenge MR_0.
- Zweitens, ob der Raum ein metrischer Raum ist, also auf seinen Elementen eine Metrik definiert ist (Definition 4.7). Ist der Raum ein Merkmalsraum, ist dies meistens der Fall.
- Drittens, ob der Raum wie in den meisten Fällen ein euklidischer Raum ist oder nicht. Ein euklidischer Raum ist mit der euklidischen Distanz naturgemäß ein metrischer Raum.
- Viertens für den Fall eines euklidischen Raumes, welche Dimension er hat beziehungsweise, ob das Ähnlichkeitsmaß beliebig hohe Dimensionen unterstützt.

Es sei angemerkt, dass man die letzten drei Unterscheidungen im Falle von Merkmalsräumen sowohl für den Raum der Primärdaten als auch für den Raum der transformierten Daten vornehmen kann.

Klassierung

Die meisten der Ähnlichkeitsmaße unserer Betrachtung werden auf den Primärdaten und im zwei- oder dreidimensionalen euklidischen Raum berechnet. Allen voran sind hier die Aggregate über synchrone Glieder und d_{E} zu nennen. Weil viele andere auf ähnliche Weise auf der euklidischen Distanz basieren, fallen sie in die gleiche Kategorie. Dazu gehören zum Beispiel die Hausdorff-Distanzen, $\mathrm{d}_{\mathrm{DTW}}$, die auf LCSS aufbauenden Unähnlichkeitsmaße und deren Abwandlungen $\mathrm{d}_{\mathrm{EDR}}$, $\mathrm{d}_{\mathrm{ERP}}$ und $\mathrm{s}_{\mathrm{Swale}}$ oder auch $\mathrm{d}_{\mathrm{DISSIM}}$.

Solche wie LCSS auf der euklidischen Distanz basierende Unähnlichkeitsmaße sind für die Verwendung in Graphen nicht geeignet [HKL05, Abschn. 2.1]. Bei d_{RN} und $\mathsf{d}_{\mathrm{Graph}}$ ist nämlich der Raum der Trajektorien euklidisch.

$\mathsf{d}_{\mathrm{AAL}}$ und $\mathsf{d}_{\mathrm{SpADe}}$ sind Paradebeispiele für die Verwendung eines Merkmalsraumes zur Berechnung der Unähnlichkeit. Der Raum der Primärdaten ist dabei ein euklidischer, weil die Strecken der Trajektorien als Vektoren eines solches betrachtet werden. Da $\mathsf{d}_{\mathrm{SBD}}$ nicht die Primärdaten, also das ursprüngliche Sampling, sondern durch ein Resampling gewonnene Trajektorien nutzt, schränken wir seine Zuordnung zu MR_0 ein. Von einem Merkmalsraum zu sprechen ist jedoch nicht treffend, weil kein Raum mit tatsächlichen Merkmalen zum Einsatz kommt. In der Berechnung von $\mathsf{d}_{\mathrm{SPM}}$ werden Positionsdistanz und Richtungsdistanz verwendet, was einen interessanten Grenzfall darstellt. Man kann diese zwar als eine Art Merkmalsraum verstehen, wir verzichten allerdings darauf, weil das vorgeschlagene Unähnlichkeitsmaß sich unmittelbar auf den Primärdaten berechnen lässt. Ähnlich verhält sich $\mathsf{d}_{\mathrm{TQuEST}}$, denn auch wenn die Berechnung nicht auf den Primärdaten, sondern den TCT stattfindet, so werden diese für jede Anfrage in Abhängigkeit der Schwelle θ generiert.

$\mathsf{d}_{\mathrm{TQuEST}}$ ist außerdem ein Beispiel für ein Unähnlichkeitsmaß, das nicht beliebig viele Dimensionen unterstützt [YCW+12]. Auch $\mathsf{d}_{\mathrm{SpADe}}$ sieht keine beliebige Dimensionalität der Positionen vor. In der Publikation, in der $\mathsf{d}_{\mathrm{AAL}}$ vorgestellt wird, handelt es sich hingegen zwar um Trajektorien im zweidimensionalen euklidischen Raum, obwohl die Technik auf Räume höherer Dimension erweitert werden kann. Dies ist auch bei vielen weiteren Ähnlichkeitsmaßen, etwa $\mathsf{d}_{\mathrm{CPD}}$, $\mathsf{d}_{\mathrm{LCSS\_1}}$ und $\mathsf{d}_{\mathrm{SBD}}$, der Fall.

6.2 Anforderungen an Zeitstempel und Länge

Klassifizierung

Die vorgestellten Ähnlichkeitsmaße haben verschiedene Anforderungen, was die Beschaffenheit der Zeitstempel angeht. Gemein ist ihnen zunächst lediglich, dass sie zwei Trajektorien konsumieren, und gemäß unserer Definition von Trajektorien (Definition 4.3), dass diese über jeweils streng monoton steigende Zeitstempel verfügen.

Für stärkere Anforderungen und insbesondere den Zusammenhang zwischen den beiden Trajektorien schlagen wir die folgenden drei Klassifikationen vor:

Definition 6.1 (Identität der Zeitstempel) *Seien* σ *und* τ *Trajektorien der Längen* n *und* m*. Die Identität der Zeitstempel (**ZS**) als Anforderung für Ähnlichkeitsmaße auf Trajektorien wird durch folgende Kategorien konkretisiert:*

- ***0**: Es muss keine Relation zwischen den Zeitstempeln der Trajektorien geben.*
- ***R**: Die ersten und letzten Zeitstempel (engl.* Range*) müssen identisch sein:* $t_{\sigma,1} = t_{\tau,1} \wedge t_{\sigma,n} = t_{\tau,m}$.
- ***I**: Alle Zeitstempel der Trajektorien müssen identisch sein:* $n = m \wedge \forall i \in \{1, \dots n\}\colon t_{\sigma,i} = t_{\tau,i}$.

Es gilt offensichtlich: $ZS_I \subseteq ZS_R$ und $ZS_R \cap ZS_0 = \emptyset$.

Definition 6.2 (Intervalle der Zeitstempel) *Seien* σ *und* τ *Trajektorien der Längen* n *beziehungsweise* m*. Für die Zeitintervalle der Trajektorien (**ZI**) unterscheiden wir zwei Kategorien:*

- ***0**: Die Zeitintervalle können beliebig sein.*
- ***Ä**: Die Zeitintervalle müssen äquidistant (Abschnitt 4.1.1) sein:* $\forall i \in \{1, \dots n-1\}\colon t_{\sigma,i+1} - t_{\sigma,i} = c$ *und* $\forall j \in \{1, \dots m-1\}\colon t_{\sigma,j+1} - t_{\sigma,j} = c'$ *für Konstanten* $c, c' \in \mathbb{R}^+$.

Es sei die Unabhängigkeit von der Klassifikation ZS betont. Auch wenn in vielen Fällen $c = c'$ gilt, so muss dies nicht zwangsweise so sein.

Definition 6.3 (Länge der Trajektorien) *Seien* σ *und* τ *Trajektorien der Längen* n *beziehungsweise* m *und* d *eine Distanzfunktion auf Positionen. Für ein Ähnlichkeitsmaß auf Trajektorien kann es folgende Kategorien bezüglich der Länge der Trajektorien (**LN**) geben:*

- ***0**: Die Länge (Definition 4.5) der Trajektorien kann beliebig sein.*
- ***L**: Die Länge der Trajektorien muss identisch sein:* $n = m$.
- ***S**: Das Akkumulat (Definition 4.5, engl.* spatial length*) der Trajektorien muss identisch sein:* $\sum_{i=1}^{n-1} d(\sigma[i], \sigma[i+1]) = \sum_{j=1}^{m-1} d(\tau[j], \tau[j+1])$.
- ***T**: Die zeitliche Länge (auch Dauer, engl.* temporal length*) der Trajektorien muss identisch sein:* $t_{\sigma,n} - t_{\sigma,1} = t_{\tau,m} - t_{\tau,1}$.

Man beachte, dass zwar $LN_0 \cap (LN_L \cup LN_S \cup LN_T) = \emptyset$ gilt, die Klassen LN_L, LN_S und LN_T aber weder paarweise disjunkt sind, noch deren symmetrische Differenz leer ist. Man beachte außerdem, dass identische Zeitstempel Trajektorien der gleichen Länge und zeitlichen Länge implizieren: $ZS_I \subseteq LN_L \cup LN_T$. Die Umkehrungen gelten beide nicht unbedingt. Selbst die striktesten bezüglich der Länge geforderten Eigenschaften können keine unterschiedlichen Zeitstempel ausschließen: $ZS_I \not\supseteq LN_L \cap LN_S \cap LN_T$. Gleichwohl

gilt: $\mathrm{ZS}_\mathrm{I} = \mathrm{ZS}_\mathsf{R} \cap \mathrm{ZI}_\ddot{\mathrm{A}} \cap \mathrm{LN}_\mathrm{L}$. Außderdem impliziert ein identischer zeitlicher Bereich auch gleiche zeitliche Länge: $\mathrm{ZS}_\mathsf{R} \subseteq \mathrm{LN}_\mathsf{T}$.

Möchte man ein Ähnlichkeitsmaß der Klasse LN_L, das also nur Trajektorien der gleichen Länge akzeptiert, auf Trajektorien mit ungleicher Länge verwenden, gibt es mehrere Möglichkeiten, dies dennoch zu tun. Ob eine, beziehungsweise welche davon sinnvoll ist, hängt natürlich von der Anwendung ab. Nehmen wir zwei Trajektorien σ und τ mit Längen n beziehungsweise m an mit $n > m$. Erstens kann man die längere der beiden Trajektorien schlicht „abschneiden“, also die Glieder von σ ab dem Index $m+1$ verwerfen [YCW+12]. Zweitens kann man eine oder beide Trajektorie(n) resamplen, also die Glieder durch neue ersetzen, was uns zurück zur in Abschnitt 5.3.12 beschriebenen Technik (Shape-Based-Distance) führt. Drittens kann man alle möglichen Subtrajektorien (Definition 4.4) von σ mit der Länge m mit τ vergleichen und auf diese Weise ein *sliding window* über σ „schieben“ [KCPM01, TPN+09].

Klassierung

Alle Aggregate über synchrone Glieder d_AI fordern identische Zeitstempel, was ihr Name schon nahelegt: $\mathsf{d}_\mathrm{AI} \in \mathrm{ZS}_\mathrm{I}$. Damit müssen die Trajektorien auch die gleiche Länge besitzen. Auch d_PF ist nur ein sinnvolles Unähnlichkeitsmaß auf Trajektorien mit identischen Zeitstempeln. Bei Aggregaten über synchrone Glieder spielt andererseits die Größe der Intervalle zwischen den Zeitstempeln keine Rolle, auch wenn sie in vielen Datensätzen äquidistant sind: $\mathsf{d}_\mathrm{AI} \in \mathrm{ZI}_0$. Das hängt damit zusammen, dass diese Unähnlichkeitsmaße unelastisch sind. Wendet man hingegen eine Technik wie DTW an, bewegen wir uns dynamisch, also in Abhängigkeit von einzelnen Positionsdistanzen, durch die Zeitdimension. Variierende Intervallgrößen bei den Zeitstempeln können dabei schnell problematisch werden. Insbesondere wenn ein *warping window* angewendet wird, das nach dem Index die Gliedpaarungen beschränkt, ist die Verwendung von d_DTW auf Daten mit nicht äquidistanten Zeitstempeln nicht mehr sinnvoll, auch wenn sie prinzipiell möglich ist. Genau entnehmen wir der Definition 5.13, dass auch LCSS durch den Parameter δ genau so eine Beschränkung durchführt. Damit sind auch die davon abgeleiteten Unähnlichkeitsmaße $\mathsf{d}_\mathrm{LCSS\_1}$, $\mathsf{d}_\mathrm{LCSS\_2}$, $\mathsf{d}_\mathrm{LCSS\_SM\_1}$ und $\mathsf{d}_\mathrm{LCSS\_SM\_2}$ auf Trajektorien mit äquidistanten Zeitstempeln deutlich besser geeignet.

Bei $\mathsf{d}_\mathrm{SBD\_TS}$ müssen die Zeitstempel der Trajektorien, auf denen die Berechnung stattfindet, äquidistant sein [YAS03, Abschn. 3.3]. Bei $\mathsf{d}_\mathrm{SBD\_S}$ hingegen soll die räumliche Länge der Strecken zwischen den Gliedern einheitlich sein. Ersteres wird durch zeitliche Normalisierung erreicht, letzteres durch räumliche Normalisierung. Das erfordert aber im ersten

Fall Trajektorien der gleichen zeitlichen Länge, im letzten ihr gleiches Akkumulat. Es gilt also $\mathsf{d}_{\text{SBD\_TS}} \in \text{LN}_\mathsf{T}$ und $\mathsf{d}_{\text{SBD\_S}} \in \text{LN}_\mathsf{S}$. Nach der Normalisierung kann man dann tatsächlich von Trajektorien der gleichen Länge ausgehen, die für die Berechnung notwendig sind. $\mathsf{d}_{\text{SBD\_TS}}$ ist dann nur sinnvoll, wenn nicht nur die Zeitintervalle äquidistant, sondern auch die Zeitstempel identisch sind. Dieses Merkmal haben wir bei $\mathsf{d}_{\text{SBD\_S}}$ nicht, da hier tatsächlich nur die Raumdimension von Interesse ist. Es ist sogar das Gegenteil der Fall: denn die Ähnlichkeit zwischen zwei Trajektorien mit äußerst unterschiedlicher zeitlicher Dauer und ähnlicher Form zeigt sich erst nach der räumlichen Normalisierung, die Teil der Berechnung von $\mathsf{d}_{\text{SBD\_S}}$ ist. Für die flächenbasierte Distanz d_{ABD} geben die Autoren keine Anforderungen an Zeitstempel und Länge der Trajektorien an. Tatsächlich ist die Verwendung mit beliebigen Trajektorien denkbar. Die Berechnung der Schnittpunkte und damit der Regionen zwischen ihnen gestaltet sich allerdings deutlich einfacher, wenn die Zeitstempel identisch sind. Ebenso ist $\mathsf{d}_{\text{DISSIM}}$ auf Trajektorien mit unterschiedlichen Zeitstempel anwendbar, wenn auf Interpolation zurückgegriffen wird. Dies ist nicht nötig, wenn die Zeitstempel identisch sind. Sind sie dies nicht, gilt es allerdings zu beachten, dass zumindest die ersten und die letzten beiden Zeitstempel identisch sein müssen, denn sonst lässt sich die Funktion $\mathsf{D}(\mathsf{t})$ nicht beschreiben: $\mathsf{d}_{\text{DISSIM}} \in \text{ZS}_\mathsf{R} \subseteq \text{LN}_\mathsf{T}$. Yang et al. schlagen mit Sequence-Pattern-Mining eine Technik vor, die Trajektorien segmentiert, bevor sie verglichen werden. Die Länge der dafür eingegebenen Trajektorien muss zwar nicht identisch sein, jedoch muss dies der Fall sein für das vorgeschlagene Unähnlichkeitsmaß, das wir untersuchen. Daher ordnen wir d_{SPM} trotzdem LN_L zu.

Für $\mathsf{d}_{\text{SpADe}}$ sind äquidistante Zeitstempel nötig, weil die Technik, um lokale Patterns zu extrahieren (*General Match*), dies verlangt. Die zeitliche Road-Network-Distanz $\mathsf{d}_{\text{RN\_T}}$ stellt keine Anforderungen an Identität und Intervallgrößen der Zeitstempel oder Länge der Trajektorien; bei ihr ist es eher umgekehrt: Die Distanz ergibt sich gerade durch die Unterschiede der Zeitstempel, zu denen beide Trajektorien eine Position passieren. Anders sieht es bei der räumlichen Road-Network-Distanz $\mathsf{d}_{\text{RN\_S}}$ aus. Die Zeitstempel müssen zwar nicht für die gesamten Trajektorien identisch sein, jedoch muss die Position zu den interessanten Zeitpunkten (ToI) für beide Trajektorien bekannt sein. Da die Autoren in dem speziellen Raum der Trajektorien keine Möglichkeiten der Interpolation angeben, kommt dies der eingeschränkten Forderung nach identischen Zeitstempeln gleich. Die Einschränkung wird insbesondere dadurch deutlich, dass solche Trajektorien deswegen nicht die gleiche Länge haben müssen, also $\mathsf{d}_{\text{RN\_S}} \in \text{LN}_0$. Die schlichte Graph-basierte Distanz $\mathsf{d}_{\text{Graph}}$ hingegen ist nur verwendbar, wenn die Trajektorien die gleiche Länge haben und nur sinnvoll, wenn die Zeitstempel identisch sind: $\mathsf{d}_{\text{Graph}} \in \text{LN}_\mathsf{L} \cap \text{ZS}_\mathsf{I}$.

6.3 Akkumulation und Elastizität

Klassifizierung

Die Ähnlichkeit zwischen zwei Trajektorien bezieht sich in aller Regel auf die Ähnlichkeit ihrer Glieder, wie bereits in Abschnitt 5.1 erwähnt. Wie aus einer solchen Menge von Werten ein einzelner gewonnen wird, variiert jedoch.

Definition 6.4 (Akkumulation der Glieder) *Ein Ähnlichkeitsmaß* $\mathtt{d}$ *auf Trajektorien, das Distanzen von Gliedern akkumuliert (**AK**), tut dies auf einer der folgenden Arten:*

- ***E**: In das Ergebnis von* $\mathtt{d}$ *fließt tatsächlich nur die Distanz eines einzelnen Gliedpaares ein.*[1]
- ***M**: In das Ergebnis von* $\mathtt{d}$ *fließen die Distanzen mehrerer Gliedpaare ein, jedoch nicht für alle Glieder beider Trajektorien.*
- ***A**: In das Ergebnis von* $\mathtt{d}$ *fließen Distanzen für alle Glieder beider Trajektorien ein.*

Die Klassen $\mathrm{AK}_\mathsf{E}, \mathrm{AK}_\mathsf{M}$ und AK_A sind paarweise disjunkt.

Es ist darüber hinaus nicht nur interessant, wie mit den Gliedpaaren verfahren wird, sondern auch, welche Gliedpaare überhaupt gebildet werden. Das führt uns zum Begriff der Elastizität, der in Abschnitt 5.3.6 schon angerissen wurde.

Definition 6.5 (Elastizität) *Seien* σ *und* τ *zwei für ein Ähnlichkeitsmaß* $\mathtt{d}$ *geeignete Trajektorien. Wir definieren die Elastizität (**EL**) wie folgt:*

- ***0**: Bei seiner Berechnung bildet* $\mathtt{d}$ *ausschließlich Paare zwischen Gliedern des gleichen Index. Es wird also niemals eine Distanz zwischen* $\sigma[\mathtt{i}]$ *und* $\tau[\mathtt{j}]$ *berechnet, wenn* $\mathtt{i} \neq \mathtt{j}$*. Wir sagen:* $\mathtt{d}$ *ist **unelastisch**.*
- ***E**: Bei der Berechnung von* $\mathtt{d}$ *werden Paare zwischen Gliedern mit ungleichem Index gebildet. Wir sagen:* $\mathtt{d}$ *ist **elastisch**.*

Wenn ein elastisches Ähnlichkeitsmaß es erlaubt, Glieder überhaupt nicht zu paaren, nennen wir es auch **überelastisch**: $\mathrm{EL}_{\ddot{\mathsf{U}}} = \overline{\mathrm{AK}_\mathsf{A}} \cap \mathrm{EL}_\mathsf{E}$.

Die Literatur kennt für elastische Ähnlichkeitsmaße auch den Begriff des *time warping* oder *time shifting* [CNOT07]. Im Gegensatz dazu steht in gewissem Sinne das *amplitude shifting*, das Verschiebungen in der Raumdimension statt in der Zeitdimension zulässt. Ähnlichkeitsmaße, die solche Verschiebungen erlauben, unterscheiden wir durch eine andere Klassifikation in Abschnitt 6.11.

[1] Natürlich ignoriert kaum ein Ähnlichkeitsmaß bestimmte Glieder vollkommen. Bei Ähnlichkeitsmaßen dieser Kategorie ist das Ergebnis vielmehr oft die Distanz genau eines Gliedpaares. Es wäre also ein Leichtes, die Trajektorien zu verändern, ohne dass sich deren Ähnlichkeit ändert.

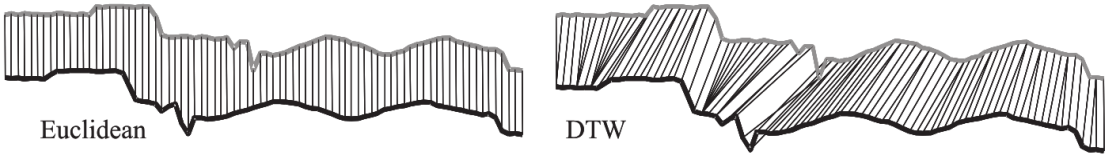


Abbildung 6.1: Elastische Ähnlichkeitsmaße erlauben es im Gegensatz zu unelastischen, Glieder mit ungleichem Index zu paaren [KR05].

Die Unelastizität eines Ähnlichkeitsmaßes impliziert, dass man es nur auf Trajektorien der gleichen Länge anwenden kann: $\mathrm{EL}_0 \subseteq \mathrm{LN}_\mathsf{L}$.[2] Im Abschnitt der Klassierung werden wir sehen, dass die Umkehrung nicht gilt.

Klassierung

Viele der zuerst in Kapitel 5 vorgestellten Ähnlichkeitsmaße lassen nur die Positionsdistanz eines einzelnen Gliedpaares in den Wert einfließen: $\mathsf{d}_{\mathrm{CPD}}, \mathsf{d}_{\mathrm{Hausdorff}}, \mathsf{d}_{\mathrm{MOHD}}, \mathsf{d}_{\mathrm{Fréchet}} \in \mathrm{AK}_\mathsf{E}$. All diese Ähnlichkeitsmaße sind auch elastisch. $\mathsf{d}_{\mathrm{CPD}}, \mathsf{d}_{\mathrm{Hausdorff}}, \mathsf{d}_{\mathrm{MOHD}}, \mathsf{d}_{\mathrm{Fréchet}} \in \mathrm{EL}_\mathsf{E}$. Es gilt aber keineswegs $\mathrm{AK}_\mathsf{E} \subseteq \mathrm{EL}_\mathsf{E}$. $\mathsf{d}_{\mathrm{AI\_min}}, \mathsf{d}_{\mathrm{AI\_max}}, \mathsf{d}_{\mathrm{AI\_median}} \in \mathrm{AK}_\mathsf{E} \cap \mathrm{EL}_0$ sind Beispiele dafür. $\mathsf{d}_{\mathrm{AI\_median}}$ ist allerdings ein Sonderfall, denn bei gerader Länge der Trajektorien müssen wir es eigentlich AK_M zuordnen. Die meisten Ähnlichkeitsmaße sind gewiss der Klasse $\overline{\mathrm{AK}_\mathsf{E}}$ zuzuordnen, berechnen sich also aus mehreren oder allen Gliedern der zu vergleichenden Trajektorien. Inbesondere sind die weit verbreiteten euklidischen Distanzen $\mathsf{d}_{\mathrm{AI\_sum}}, \mathsf{d}_{\mathrm{AI\_mean}}, \mathsf{d}_{\mathrm{AI\_RMS}}, \mathsf{d}_{\mathrm{E}}$ sowie $\mathsf{d}_{\mathrm{DTW}}, \mathsf{d}_{\mathrm{EDR}}, \mathsf{d}_{\mathrm{ERP}}$ und alle Ähnlichkeitsmaße der Art $\mathsf{d}_{\mathrm{LCSS}}$ zu nennen. Während die Aggregate über synchrone Glieder unelastisch sind, sind alle anderen genannten Ähnlichkeitsmaße elastisch, die auf LCSS basierenden sogar überelastisch. Dazu bemerken Vlachos et al. den Vorteil bei unterschiedlichen Zeitstempeln:

> „The LCSS model allows stretching and displacement in time, so we can detect similarities in movements that happen with different speeds, or at different times." [VKG02]

Die Aggregate über Punkt-Trajektorien-Distanzen d_{AP} sind genau wie die Closest-Pair-Distance $\mathsf{d}_{\mathrm{CPD}}$ elastisch, unterscheiden sich von ihr aber dadurch, dass sie AK_A und nicht AK_E angehören. Ansonsten sind sie sich aufgrund ihrer offensichtlichen Verwandtschaft äußerst ähnlich, wie wir der Übersichtstabelle 7.2 entnehmen können. Das Unähnlichkeitsmaß d_{PF} ist ein außergewöhnlicher Fall, weil es zwar in der Klasse LN_L, aber elastisch

[2] Das gilt nur, wenn man von Techniken, die ein Resampling der Trajektorien verwenden, absieht.

ist, also trotzdem Glieder mit unterschiedlichem Index miteinander vergleichen kann. Abgesehen von dieser Ausnahme gilt in der Praxis: $LN_L \subseteq EL_0$. Ebenfalls interessant ist d_{SBD}. Die Definition selbst vergleicht ausschließlich Glieder des gleichen Index, ist also unelastisch. Dennoch kann man das Unähnlichkeitsmaß als elastisch ansehen, da die Berechnung ja auf den normalisierten Trajektorien durchgeführt wird, also faktisch Glieder mit unterschiedlichen Indizes der ursprünglichen Trajektorien verglichen werden.

Bei einigen Unähnlichkeitsmaßen ist die Einordnung in Klassen für Elastizität nur bedingt sinnvoll, weil sie in einem Merkmalsraum berechnet werden. Das betrifft etwa d_{AAL} und d_{SpADe}. Es gibt auch Ähnlichkeitsmaße, die ohne Merkmalsraum auskommen und sich trotzdem nicht durch Paarungen von Gliedern berechnen, so etwa die flächenbasierten Ähnlichkeitsmaße d_{ABD} und d_{DISSIM}. Auch bei ihnen kategorisieren wir nicht nach Elastizität.

Die zeitliche Road-Network-Distanz d_{RN_T} ordnen wir der Klasse AK_E zu, da Hwang et al. scheinbar voraussetzen, dass eine Position innerhalb einer Trajektorie nicht öfter als einmal vorkommt. Wir tun dies unter Vorbehalt, da die Frage, wie die zeitliche Distanz in anderen Fällen berechnet werden soll, offen ist. Die räumliche Road-Network-Distanz d_{RN_S} können wir allerdings problemlos der Klasse AK_M zuordnen. Das impliziert dieselbe Klasse für die spatiotemporale Road-Network-Distanz: $d_{RN\_S}, d_{RN\_TS} \in AK_M$. Alle drei Distanzen können wir als elastisch betrachten, weil die Glieder ausschließlich in Abhängigkeit der ToI beziehungsweise PoI ausgewählt werden und infolgedessen Glieder mit unterschiedlichen Indizes miteinander verglichen werden. An der Definition der Graphbasierten Distanz d_{Graph} können wir leicht ablesen, dass sie den Klassen AK_A und EL_0 angehört. Sie fordert allerdings auch identische Zeitstempel ($d_{Graph} \in ZS_I$).

6.4 Längen- und Zeitempfindlichkeit

Klassifizierung

Das Ergebnis mancher Unähnlichkeitsmaße hängt stark von der Länge der Trajektorien ab. Damit ist gemeint, dass ihre Unähnlichkeit im Allgemeinen größer wird, je länger die verglichenen Trajektorien werden. Für eine mathematischere Definition benutzen wir den Begriff der *Verlängerung* einer Trajektorie: Eine um a Glieder **verlängerte Trajektorie** einer Trajektorie $\tau = \big(\langle p_1, t_1\rangle, \ldots \langle p_n, t_n\rangle\big)$ ist eine Trajektorie mit echt größerer Länge $n + a$ ($a > 0$) und bis zum einschließlich n. Glied identischen Gliedern: $\tau' = \big(\langle p_1, t_1\rangle, \ldots \langle p_n, t_n\rangle, \langle p_{n+1}, t_{n+1}\rangle, \ldots \langle p_{n+a}, t_{n+a}\rangle\big)$.

Definition 6.6 (Längenempfindlichkeit) *Seien* σ *und* τ *zwei für ein Unähnlichkeitsmaß* d *geeignete Trajektorien der Längen* n *beziehungsweise* m*. Wir definieren Längenempfindlichkeit (**LE**) wie folgt:*

- ***L**: $d \in AK_A$ und es existieren keine Verlängerungen σ' von σ und τ' von τ, sodass gilt: $d(\sigma', \tau') < d(\sigma, \tau)$. Wir sagen: d ist **längenempfindlich**.*
- ***0**: Ist d nicht längenempfindlich, nennen wir es **längenunempfindlich**.*

Per Definition ist $LE_L \subseteq AK_A$, da diese Klassifikation nur sinnvoll ist, wenn tatsächlich alle Glieder in die Berechnung mit einfließen. Andernfalls handelt es sich automatisch um ein längenunempfindliches Ähnlichkeitsmaß: $AK_E \cup AK_M \subseteq LE_0$. Längenunempfindliche Ähnlichkeitsmaße der Klasse AK_A normalisieren ihren Wert in der Regel mit der Länge der Trajektorien. Man kann Unähnlichkeitsmaße, die längenempfindlich sind, zumeist längenunempfindlich machen, indem man das Ergebnis in Relation zu der Länge der eingegebenen Trajektorien setzt.

So wie ein Ähnlichkeitsmaß empfindlich auf die Länge der Trajektorien reagieren kann, können auch die Zeitstempel sein Ergebnis maßgeblich beeinflussen oder nicht. Wir führen dafür eine neue Klassifikation ein.

Definition 6.7 (Zeitempfindlichkeit) *Seien* σ *und* $\tau = \big(\langle p_1, t_1\rangle, \dots \langle p_n, t_n\rangle\big)$ *zwei für ein Ähnlichkeitsmaß* d *geeignete Trajektorien. Für die Zeitempfindlichkeit (**ZE**) von* d *unterscheiden wir folgende Klassen:*

- ***Z**: Es existieren streng monoton steigende Zeitstempel $t'_1 < \dots t'_n$, sodass gilt: $d(\sigma, \big(\langle p_1, t'_1\rangle, \dots \langle p_n, t'_n\rangle\big)) \neq d(\sigma, \tau)$. Wir sagen: d ist **zeitempfindlich**.*
- ***P**: d ist nicht zeitempfindlich, aber es existiert eine Permutation der Positionen von τ, sodass gilt:*
  *$d(\sigma, \big(\langle p_i, t_1\rangle, \dots \langle p_j, t_n\rangle\big)) \neq d(\sigma, \tau)$. Wir sagen: d ist **permutationsempfindlich**.*
- ***0**: d ist weder zeitempfindlich noch permutationsempfindlich. Wir sagen: d ist **zeitunempfindlich**.*

Während zeitempfindliche Ähnlichkeitsmaße ihren Wert bei der Änderung der Zeitstempel einer Trajektorie potentiell verändern, ignorieren zeitunempfindliche Ähnlichkeitsmaße die Zeitstempel gänzlich. Bei einem permutationsempfindlichen Ähnlichkeitsmaß fließt die Zeit implizit ein, denn die Reihenfolge der Glieder hat Einfluss auf sein Ergebnis.

Klassierung

Interessanterweise unterscheiden sich die euklidischen Unähnlichkeitsmaße in der Klassifikation der Längenempfindlichkeit. Während d_{AI_sum} und d_E umso größer werden, je

länger die Trajektorien sind, enthalten $\mathsf{d}_{\text{AI\_mean}}$ und $\mathsf{d}_{\text{AI\_RMS}}$ den Bruch $\frac{1}{n}$ und sind somit normalisiert. Die meisten Ähnlichkeitsmaße sind längenunempfindlich. Zu längenempfindlichen Ähnlichkeitsmaßen zählen dagegen d_{DTW}, d_{EDR}, d_{ERP}, $\mathsf{s}_{\text{Swale}}$ sowie d_{ABD} und $\mathsf{d}_{\text{DISSIM}}$. Das zusammengesetzte Unähnlichkeitsmaß d_{SPM} ist nicht längenempfindlich, weil keiner der drei Summanden längenempfindlich ist: $\mathsf{d}_{\text{AI\_RMS}}$ ist es nicht, die Positionsdistanz ist nicht von der Länge abhängig und die Richtungsdistanz ist mit der Länge der Trajektorien normalisiert.

Wir können auch Ähnlichkeitsmaße, die einen Merkmalsraum verwenden, bezüglich ihrer Längenempfindlichkeit klassieren. Beispielsweise ist $\mathsf{d}_{\text{SpADe}}$ längenempfindlich, da mit steigender Anzahl der Glieder auch die Anzahl der lokalen Patterns wächst und somit ebenfalls die Größe der Matching-Matrix und der kürzeste Pfad durch sie. Auch d_{AAL} ist längenempfindlich, weil DTW längenempfindlich ist und die Länge der AAL-Repräsentation einer Trajektorie linear mit ihrer Länge wächst.

Alle Aggregate über synchrone Glieder und auch d_{E} sind permutationsempfindlich, also Elemente der Klasse ZE_{P}, denn sie bilden Paare zwischen Glieder mit gleichem Index, die sich durch Permutation der Positionen ändern würden. Die Werte der Zeitstempel fließen allerdings nicht ein. $\mathsf{d}_{\text{CPD}} \in \text{AK}_{\mathsf{E}}$ gehört zur Klasse ZE_0, ist also zeitunempfindlich. Auch die Hausdorff-Distanz ist zeitunempfindlich: $\mathsf{d}_{\text{Hausdorff}} \in \text{ZE}_0$. Diese Klassenzugehörigkeit können wir als wichtigsten Unterschied zur Fréchet-Distanz verstehen, die ja – wie bei deren Definition erläutert – das „Problem“ aus Abbildung 5.1 löst, indem sie die Reihenfolge der Glieder berücksichtigt: $\mathsf{d}_{\text{Fréchet}} \in \text{ZE}_{\mathsf{P}}$. Die modifizierte Hausdorff-Distanz gehört auch zu dieser Klasse: $\mathsf{d}_{\text{MOHD}} \in \text{ZE}_{\mathsf{P}}$. Die „klassischen“ elastischen Ähnlichkeitsmaße sind ebenfalls permutationsempfindlich, denn wie wir an DTW sehen (Definition 5.12), müssen die Trajektorien durch tail in etwa gleichmäßig traversiert werden, damit die Unähnlichkeit klein bleibt. Die Zeitstempel selbst werden allerdings nicht berücksichtigt. Das gilt ebenso für d_{LCSS}, d_{EDR}, d_{ERP} und $\mathsf{s}_{\text{Swale}}$. Der Großteil der Unähnlichkeitsmaße gehört damit der Klasse ZE_{P} an und lässt die Werte der Zeitstempel nicht einfließen. Diese Erkenntnis deckt sich mit der Literatur [FGT07, HKL05]. Eine Form der Zeitempfindlichkeit ($\overline{\text{ZE}_0}$) – insbesondere oft die letzte – liegt fast immer vor, wenn mehrere Glieder zu dem Ergebnis beitragen ($\overline{\text{AK}_{\mathsf{E}}}$). Die Ausnahmen bilden dabei Aggregate über Punkt-Trajektorien-Distanzen und die verwandte Grid-basierte Distanz.

Die Shape-Based-Distance beruht auf einer Normalisierung der Trajektorien. Dadurch reagiert sie im Gegensatz zu den meisten anderen Ähnlichkeitsmaßen auf Änderungen der Zeitstempel und ist damit Element von ZE_{Z}. Das gilt sowohl für $\mathsf{d}_{\text{SBD\_TS}}$ als auch für $\mathsf{d}_{\text{SBD\_S}}$, weil auch bei letzterer die normalisierte Trajektorie andere Positionen trägt.

Fließt die Zeit in die Berechnung der Ähnlichkeit zweier Trajektorien ein, geschieht dies zumeist indirekt. Bei DISSIM zum Beispiel, indem die Interpolation für die Position zu einem bestimmtem Zeitpunkt einen anderen Wert ergibt, wenn man die Zeitstempel der Glieder ändert. Es gilt also: $\mathtt{d}_{\mathrm{DISSIM}} \in \mathrm{ZE}_{\mathtt{Z}}$. Das unterscheidet DISSIM von der anderen vorgestellten flächenbasierten Distanz, denn $\mathtt{d}_{\mathrm{ABD}} \in \mathrm{ZE}_{\mathtt{P}}$. Die reine Fläche ist nämlich, wie die Autoren selbst anmerken, ein zeitunabhängiger Wert [NB03, Abschn. 2.6]. Bei $\mathtt{d}_{\mathrm{RN\_T}}$ allerdings fließt die Zeit tatsächlich direkt ein, denn diese Unähnlichkeit ergibt sich aus der Distanz von Zeitstempeln. Das Pendant $\mathtt{d}_{\mathrm{RN\_S}}$ verhält sich wiederum ähnlich wie DISSIM und betrachtet die Trajektorie als Funktion der Zeit. Die Threshold-Distanz $\mathtt{d}_{\mathrm{TQuEST}}$ ist zeitempfindlich, weil sie anhand der TCT berechnet wird, die von den Zeitstempeln der Zeitreihen abhängen.

6.5 Maßdimension

Klassifizierung

Von den in Kapitel 5 vorgestellten Techniken ist LCSS die erste, die eine Art Quantisierung vornimmt. Das Ergebnis ist keine Akkumulation von Punktdistanzen – also kein geometrisch sinnvoll zu interpretierender Wert, sondern eine Anzahl von Gliedern, die bestimmte Kriterien erfüllen – also ein kombinatorisch ermittelter Wert. Dieser grundlegende Unterschied ist in der Literatur bekannt: Morse et al. bezeichnen die erste Klasse von Ähnlichkeitsmaßen als „based on the L1 and L2 norms“[3], die zweite als „based on a matching threshold“ [MP07, Kap. 1]. Weil bei Ähnlichkeitsmaßen der ersten Klasse tatsächlich die Distanzen zwischen Gliedern gemessen werden und Teil des Ergebnisses sind, und sie bei Ähnlichkeitsmaßen der zweiten Klasse lediglich gezählt werden, kann man die Unterscheidung durch die zugrunde liegenden Techniken *Messen* und *Zählen* treffen. Wir abstrahieren diese Idee und sprechen von der *Maßdimension* des Ähnlichkeitsmaßes, denn Zählen ist eigentlich eine spezielle Art zu messen [Hau].

Definition 6.8 (Maßdimension) *Einem Ähnlichkeitsmaß* $\mathtt{d}$ *auf Trajektorien ordnen in Abhängigkeit von seiner Maßdimension (**MD**) einer der folgenden Kategorien zu:*

- ***0**: Das Ergebnis von* $\mathtt{d}$ *hat keine sinnvolle (geometrische) Interpretation im Raum der Trajektorien.*
- ***1**: Die Einheit des Ergebnisses von* $\mathtt{d}$ *ist genau die Einheit der Positionsdistanzen im Raum der Trajektorien.*

[3] Dies ist kein glücklich gewählter Ausdruck, da ja auch LCSS für die Punktdistanzen eine L_p-Norm bemüht.

- *2: Die Einheit des Ergebnisses von* $\mathtt{d}$ *ist eine komplexere Variation der Einheit der Positionsdistanzen im Raum der Trajektorien.*

Die Klassen $\mathrm{MD}_0, \mathrm{MD}_1$ und MD_2 sind paarweise disjunkt.

Klassierung

Unähnlichkeitsmaße der Klasse MD_1 sind gewissermaßen die natürlichsten. In diese Klasse fallen unter anderem alle Unähnlichkeitsmaße, die mittels Aggregaten über synchronen Gliedern oder PTD erzeugt werden, $\mathtt{d}_{\mathrm{E}}$, die Hausdorff- und Fréchet-Distanzen, sowie $\mathtt{d}_{\mathrm{DTW}}$. Wie schon angedeutet, gehören die auf LCSS basierenden Ähnlichkeitsmaße $\mathtt{d}_{\mathrm{LCSS\_1}}, \mathtt{d}_{\mathrm{LCSS\_1}}$ und $\mathtt{s}_{\mathrm{Swale}}$ zur Klasse MD_0. So ist auch $\mathtt{d}_{\mathrm{EDR}} \in \mathrm{MD}_0$, aber $\mathtt{d}_{\mathrm{ERP}} \in \mathrm{MD}_1$. Ähnlichkeitsmaße, die die Fläche zwischen Trajektorien berechnen, gehören der Klasse MD_2 an. Von den am Anfang des Kapitels aufgelisteten Ähnlichkeitsmaßen betrifft dies $\mathtt{d}_{\mathrm{ABD}}$ und $\mathtt{d}_{\mathrm{DISSIM}}$.

Ein interessanter Fall ist $\mathtt{d}_{\mathrm{SPM}}$. Teil des Wertes ist zwar eine euklidische Distanz, die uns die Maßdimension 1 nahelegt, jedoch lässt die Summe aus dieser Distanz, der Positions- und Richtungsdistanz keine sinnvolle geometrische Interpretation zu. Wir klassieren das Unähnlichkeitsmaß daher als Element von MD_0.

Weiterhin lässt sich beobachten, dass Ähnlichkeitsmaße, die in Merkmalsräumen berechnet werden, jene diese oftmals so gestalten, dass keine geometrische Interpretation sinnvoll ist: $\mathrm{MR}_{\mathcal{M}} \subseteq \mathrm{MD}_0$. Da diese Eigenschaft in der Regel Zweck des Merkmalsraumes ist, verwundert diese Mengenbeziehung nicht.

Bei der Graph-basierten Distanz $\mathtt{d}_{\mathrm{Graph}}$ handelt es nicht um einen euklidischen Raum der Trajektorien. Sie ist ein gutes Beispiel dafür, dass die Maßdimension trotzdem 1 sein kann, denn die Distanz berechnet sich durch die Kostenfunktion im Graphen und benutzt damit ein Maß, das genau dem der Distanz für einzelne Positionen – nämlich Knoten des Graphen – entspricht. Die Road-Network-Distanzen sind hingegen kein eindeutiger Fall. Die räumliche Road-Network-Distanz ist eine Summe aus Positionsdistanzen und hat daher die Maßdimension 1. Separiert können wir auch die zeitliche Road-Network-Distanz der gleichen Klasse MD_1 zuordnen, denn Potenzierung und Wurzel heben sich auf, sodass eine Summe aus zeitlichen Differenzen übrig bleibt; mit der gleichen Einheit wie die zeitliche Distanz zwischen zwei einzelnen Gliedern, auch wenn die räumliche Komponente fehlt. Bei der spatiotemporalen Road-Network-Distanz fehlt allerdings die sinnvolle Interpretation im Raum der Trajektorien, weil zwei Werte unterschiedlicher Einheit addiert werden.

6.6 Parametrierbarkeit

Klassifizierung

Ein naheliegendes Charakteristikum ist die Parametrierbarkeit eines Ähnlichkeitsmaßes. Es gibt zahlreiche Ähnlichkeitsmaße, deren Ergebnis von weiteren Parametern neben den eingegebenen Trajektorien abhängt. Ein klarer Vorteil eines parametrierbaren Ähnlichkeitsmaßes besteht darin, besseren Einfluss auf sein Ergebnis zu nehmen und es so potentiell an die Anwendung anzupassen. Der Nachteil, den dies mit sich bringt, ist die Notwendigkeit der Belegung ebenjener Parameter. In vielen Fällen ist es schwer, die richtigen Belegungen zu finden, damit das Ergebnis den Anforderungen der Anwendung genügt. Mit steigender Anzahl der Parameter gelangt man schnell an ein ernstzunehmendes Optimierungsproblem, da der Suchraum mit den Freiheitsgraden superlinear wächst.

Wir wollen Ähnlichkeitsmaße unterscheiden, indem wir sowohl die Existenz von Parametern als auch ihre Art untersuchen.

Definition 6.9 (Parametrierbarkeit) *Sei* d *ein Ähnlichkeitsmaß und* σ *und* τ *zwei dafür geeignete Trajektorien. Die Parametrierbarkeit (**PRM**) von* d *klassifizieren wir wie folgt:*

- ***0**: Neben* σ *und* τ *konsumiert* d *keine weiteren Parameter.*
- ***A**:* d *konsumiert mindestens einen Parameter, der eindeutig der Anfrage (engl.* query*) und nicht der Berechnung des Ergebnisses zuzuordnen ist.*
- ***W**:* d *konsumiert mindestens einen Parameter, der manipuliert, welche Glieder der Trajektorien gepaart werden (engl.* warping*).*
- ***P**:* d *konsumiert mindestens eine Paarungsschwelle, also einen Wert, der als Grenzwert für die Distanz zweier Glieder fungiert.*
- ***V**:* d *konsumiert mindestens einen anderen Parameter.*

Die Klasse PRM_0 schließt andere Klassen der Parametrierbarkeit aus:
$\mathrm{PRM}_0 \cap (\mathrm{PRM}_\mathsf{A} \cup \mathrm{PRM}_\mathsf{W} \cup \mathrm{PRM}_\mathsf{P} \cup \mathrm{PRM}_\mathsf{V}) = \emptyset$. Alle anderen Klassen sind aber ausdrücklich nicht paarweise disjunkt.

Ein Ähnlichkeitsmaß, das eine Einstellung darüber erlaubt, welche Glieder miteinander gepaart werden, ist zwangsweise elastisch: $\mathrm{PRM}_\mathsf{W} \subseteq \mathrm{EL}_\mathsf{E}$. Ein zählendes Ähnlichkeitsmaß (Maßdimension 0) im Raum der Trajektorien erlaubt in aller Regel – weil sonst wenig sinnvoll – eine Einstellung der Paarungsschwelle für Gliedpaare, die gezählt werden sollen. Andersherum sind uns keine Ähnlichkeitsmaße bekannt, die Paarungsschwellen konsumieren, aber nicht zählend sind, auch wenn das durchaus denkbar wäre. In der Pra-

xis beobachtet man eine große Übereinstimmung der Mengen $\mathrm{PRM}_{\mathsf{P}} \cup \mathrm{MR}_0$ und MD_0. Zu dieser Regel gibt es allerdings auch Ausnahmen. Das Unähnlichkeitsmaß $\mathsf{d}_{\mathrm{SPM}}$ etwa konsumiert keine Paarungsschwelle, weil es nicht zählend ist, gehört aber trotzdem – wie im vorigen Abschnitt dargelegt – der Klasse MD_0 an.

Klassierung

Die Closest-Pair-Distance, alle Aggregate über synchrone Glieder oder Punkt-Trajektorien-Distanzen, d_{E}, die Hausdorff- und die Fréchet-Distanz sind parameterfrei, also Elemente der Klasse PRM_0. Elastische Ähnlichkeitsmaße (EL_{E}) der Klasse $\overline{\mathrm{AK}_{\mathsf{E}}}$ haben oft einen Parameter, um das *warping* zu beschränken: $\mathsf{d}_{\mathrm{MOHD}}, \mathsf{d}_{\mathrm{LCSS}}, \mathsf{d}_{\mathrm{PF}} \in \mathrm{PRM}_{\mathsf{W}}$. Die PF-Distanz fällt dabei ebenfalls in diese Klasse, weil δ faktisch nichts anderes ist als die Größe eines *warping windows*. Es fällt auf, dass $\mathsf{d}_{\mathrm{DTW}}$ selbst nach Definition 5.12 keinen solchen Parameter konsumiert. Wie in Abschnitt 5.3.6 erklärt, ist es jedoch gängige Praxis, ein *warping window* einzusetzen. Da dies das klassische Beispiel für einen solchen Parameter ist, führen wir also trotzdem folgende eingeschränkte Zuordnung durch: $\mathsf{d}_{\mathrm{DTW}} \in \mathrm{PRM}_{\mathsf{W}}$. Im starken Kontrast zur ursprünglichen Hausdorff-Distanz muss man bei der modifizierten einige Parameter einstellen, nämlich α, C und N. Die Funktion N entspricht dabei in gewisser Weise auch der Größe eines *warping windows*.

Die zählenden Ähnlichkeitsmaße konsumieren in der Regel eine Paarungsschwelle: $\mathsf{d}_{\mathrm{LCSS\_1}}$, $\mathsf{d}_{\mathrm{LCSS\_2}}, \mathsf{d}_{\mathrm{EDR}}, \mathsf{s}_{\mathrm{Swale}} \in \mathrm{PRM}_{\mathsf{P}}$. Das Ergebnis von LCSS hängt in großem Maße von dem Parameter ϵ, der Paarungsschwelle, ab. Dass diese nicht dynamisch geändert werden kann, bringt ihr Kritik ein [TPN+09]. Die Verbesserung mit der Sigmoidfunktion (Abschnitt 5.3.7) hat einen solchen problematischen Parameter nicht. Sie ersetzt ihn aber durch einige andere Parameter, die diese Paarungsschwelle eigentlich simulieren: $\mathsf{d}_{\mathrm{LCSS\_SM\_1}}, \mathsf{d}_{\mathrm{LCSS\_SM\_2}} \in \mathrm{PRM}_{\mathsf{V}}$. Tatsächlich erzielen diese Ähnlichkeitsmaße mit der Sigmoidfunktion mindestens genauso gute Ergebnisse wie die ursprünglichen auf LCSS basierenden, selbst wenn man eine optimale Paarungsschwelle ϵ wählt [VGK02, Kap. 3]. Allerdings müssen auch diese neuen Parameter geeignet belegt werden. Die Abstraktion von LCSS und EDR $\mathsf{s}_{\mathrm{Swale}}$ verlangt neben der Paarungsschwelle (engl. *matching threshold*) auch die Werte, die bei Paarungen (engl. *matching reward*) oder Nicht-Paarungen (engl. *gap penalty*) in das Ergebnis einfließen [WMD+13, Kap. 2]. Also gilt: $\mathsf{s}_{\mathrm{Swale}} \in \mathrm{PRM}_{\mathsf{P}} \cap \mathrm{PRM}_{\mathsf{V}}$.

Weitere Beispiele für die Klasse $\mathrm{PRM}_{\mathsf{V}}$ sind $\mathsf{d}_{\mathrm{SBD\_TS}}$ und $\mathsf{d}_{\mathrm{SBD\_S}}$, denn die Distanzberechnung benötigt zwar keine Parameter, jedoch die vorangehende Normalisierung: Bei zeitlicher Normalisierung muss das Zeitintervall $\Delta\mathsf{t}$, bei räumlicher die Streckenlänge δ

festgelegt werden. Ein weiteres Beispiel ist $\mathsf{d}_{\mathrm{SPM}}$, denn hier müssen die Gewichte der Teildistanzen belegt werden. Für die Spatial-Assembling-Distance $\mathsf{d}_{\mathrm{SpADe}}$ muss man sehr viele Parameter einstellen (*temporal scale factor*, *amplitude scale factor*, *pattern length*, *sliding step size*). Im Vergleich zu anderen Techniken scheinen passende Parameter besonders schwer zu finden zu sein [WMD+13, Abschn. 4.3]. Auch wenn einer dieser Parameter ϵ eine Paarungsschwelle ist, ordnen wir $\mathsf{d}_{\mathrm{SpADe}}$ nicht der Klasse $\mathrm{PRM}_{\mathsf{P}}$ zu, weil sich diese nicht auf die Distanz von Gliedern der Trajektorien bezieht, sondern auf lokale Patterns.

$\mathsf{d}_{\mathrm{TQuEST}}$ und alle Road-Network-Distanzen $\mathsf{d}_{\mathrm{RN\_T}}$, $\mathsf{d}_{\mathrm{RN\_S}}$ und $\mathsf{d}_{\mathrm{RN\_TS}}$ gehören der Klasse $\mathrm{PRM}_{\mathsf{A}}$ an. Erstere, weil die Schwelle θ zur Anfrage gehört, letztere, weil die PoI beziehungsweise RoI belegt werden müssen. Die Autoren der Graph-basierten Distanz $\mathsf{d}_{\mathrm{Graph}} \in \mathrm{PRM}_0$ kritisieren dies [TPN+09, Kap. 2].

6.7 Metrische Eigenschaften

Klassifizierung

Zu den wichtigen Charakteristika eines Ähnlichkeitsmaßes gehören seine metrischen Eigenschaften, also ob oder inwieweit die Kriterien aus Definition 4.7 erfüllt werden. Im positiven Fall können wir tatsächlich von einem Unähnlichkeitsmaß statt einem Ähnlichkeitsmaß sprechen, da jedes andere Ähnlichkeitsmaß die positive Definitheit verletzen würde. Für Ähnlichkeitsmaße, die ihrerseits ein Unähnlichkeitsmaß auf Positionen bemühen, setzen wir dessen metrische Eigenschaften voraus.

Definition 6.10 (Metrische Eigenschaften) *Sei* d *ein Ähnlichkeitsmaß auf Trajektorien. Wir unterscheiden bezüglich der metrischen Eigenschaften (**MTR**) folgende Kategorien:*

- ***M**:* d *ist eine Metrik gemäß Definition 4.7, wobei zwei Trajektorien auch als gleich angesehen werden, wenn nur die Positionen der Glieder identisch sind und nicht auch die Zeitstempel.*
- ***Q**:* d *ist eine Quasimetrik, aber keine Metrik.*
- ***S**:* d *ist eine Semimetrik, aber keine Metrik.*
- ***P**:* d *ist positiv definit, aber weder Quasimetrik noch Semimetrik.*
- ***0**:* d *ist nicht positiv definit.*

Die Kategorien sind so gewählt, dass ihre Klassen paarweise disjunkt sind. Wir entspannen das Kriterium der Gleichheit für das Identitätsprinzip, weil sonst alle nicht zeitempfindlichen Unähnlichkeitsmaße jegliche metrische Eigenschaft verlören: $\overline{\mathrm{ZE}_{\mathsf{Z}}} \not\subseteq \mathrm{MTR}_0$. Selbst

die sogenannten euklidischen Distanzen würden, weil die Werte der Zeitstempel nicht einfließen, das Identitätsprinzip verletzen.

Die Erfüllung der metrischen Eigenschaften, insbesondere der Dreiecksungleichung, ist für ein Unähnlichkeitsmaß zum Beispiel bei der Erstellung von klassischen Indexstrukturen in Datenbanken von äußerster Wichtigkeit, weil dann effizient große Datenmengen als Ergebnis für eine Anfrage ausgeschlossen werden können [Che05, Abschn. 6.1].

Klassierung

Die euklidischen Distanzen sind Metriken, weil sie auf der L_2-Norm basieren. Für $p < 1$ sind solche Abstände jedoch nicht metrisch [JWG00, Abschn. 2.1]. Die beiden Unähnlichkeitsmaße d_{SBD_S} und d_{SBD_TS} sind beide metrisch, weil sie diese Eigenschaft von den euklidischen Distanzen erben.

d_{CPD} erfüllt zwar Nicht-Negativität und Symmetrie, verletzt aber das Identitätsprinzip und die Dreiecksungleichung. Das gilt auch für d_{AI_min}. Daher sind $d_{CPD}, d_{AI\_min} \in MTR_0$. Die Verletzungen des Identitätsprinzips sowie der Dreiecksungleichung lassen sich durch folgendes Gegenbeispiel mit Trajektorien im eindimensionalen euklidischen Raum beweisen: Seien $\sigma_1 = \big(\langle 1,1\rangle, \langle 1,2\rangle\big)$, $\sigma_2 = \big(\langle 1,1\rangle, \langle 5,2\rangle\big)$ und $\sigma_3 = \big(\langle 5,1\rangle, \langle 5,2\rangle\big)$. Dann gilt: $d_{CPD}(\sigma_1,\sigma_2) = d_{AI\_min}(\sigma_1,\sigma_2) = 0$, obwohl $\sigma_1 \neq \sigma_2$ ist. Außerdem gilt: $d_{CPD}(\sigma_2,\sigma_3) = d_{AI\_min}(\sigma_2,\sigma_3) = 0$ sowie $d_{CPD}(\sigma_1,\sigma_3) = d_{AI\_min}(\sigma_1,\sigma_3) = 4$. Damit ist $d_{CPD}(\sigma_1,\sigma_3) > d_{CPD}(\sigma_1,\sigma_2) + d_{CPD}(\sigma_2,\sigma_3)$ und $d_{AI\_min}(\sigma_1,\sigma_3) > d_{AI\_min}(\sigma_1,\sigma_2) + d_{AI\_min}(\sigma_2,\sigma_3)$.

d_{CPD} und d_{AI_min} gehören beide der Klasse AK_E an. Es gilt jedoch keineswegs $AK_E \subseteq MTR_0$, denn d_{AI_max} und $d_{Hausdorff}$ aus derselben Klasse verletzen das Identitätsprinzip nicht.

Die Fréchet-Distanz ist tatsächlich eine Metrik [EM94]: $d_{Fréchet} \in MTR_M$. Selbiges gilt für die Hausdorff-Distanz: $d_{Hausdorff} \in MTR_M$. Die modifizierte Hausdorff-Distanz d_{MOHD} ist ein recht interessanter Fall. Im Allgemeinen verletzt sie das Identitätsprinzip, weil der Parameter α so klein gewählt werden kann, dass zwei ähnliche aber nicht identische Trajektorien eine Unähnlichkeit von 0 haben. In unserem Beispiel etwa ist $d_{MOHD}(\sigma_1,\sigma_2,0.4,N,C) = 0$. Damit ist sie nicht metrisch und Element der Klasse MTR_0. Für geeignete α jedoch, zum Beispiel $\alpha = 1$, trifft das nicht zu. Es gibt auch andere Variationen der Hausdorff-Distanz [HKR93], die die Dreiecksungleichung verletzen [CÖO05, Abschn. 4] und damit nur eine Semimetrik sind.

Auf ähnliche Weise wie $\mathrm{d}_{\mathrm{CPD}}$ im obigen Beispiel verletzt $\mathrm{d}_{\mathrm{DTW}}$ die Dreiecksungleichung [Che05, Abschn. 2.2]. Dies liegt daran, dass ein und dasselbe Glied einer Trajektorie mehrfach in das Ergebnis einfließt [CN04b, Abschn. 3.2]. Vidal et al. stellen jedoch in [VCB+88] fest, dass DTW in der Praxis nicht weit davon entfernt ist, die Dreiecksungleichung zu erfüllen.

Für die auf LCSS basierenden Unähnlichkeitsmaße gilt ebenfalls, dass sie nicht metrisch sind. Hier liegt das Problem jedoch in der Eigenschaft, dass sie zählend sind – also Maßdimension 0 haben – und eine Paarungsschwelle verwenden. Das hat zur Folge, dass das Identitätsprinzip verletzt wird, weil zwei Trajektorien, deren Glieder stets Distanzen unterhalb dieser Schwelle haben, als gleich angesehen werden, auch wenn sie es nicht sind. Dies gilt auch für $\mathrm{d}_{\mathrm{EDR}}$. Im Gegensatz dazu wurde $\mathrm{d}_{\mathrm{ERP}}$ genau dafür entworfen, metrische Eigenschaften zu haben. Dies wird dadurch erreicht, dass tatsächlich Positionsdistanzen einfließen, auch für solche Glieder, die nicht gepaart werden können [CN04b].

$\mathrm{s}_{\mathrm{Swale}}$ ist das einzige Ähnlichkeitsmaß in unserer Liste, das die Nicht-Negativität verletzt. Es ist daher nicht positiv definit und somit auch nicht metrisch. d_{PF} ist das einzige Ähnlichkeitsmaß, das deswegen nicht metrisch ist, weil es die Eigenschaft der Symmetrie nicht erfüllt. Das ist ungewöhnlich, da die Symmetrie künstlich auch für nicht symmetrische Unähnlichkeitsmaße leicht zu erzwingen ist, wie zum Beispiel bei der Hausdorff-Distanz (Abschnitt 5.3.4). Würde die PF-Distanz die Dreiecksungleichung erfüllen, wäre sie demzufolge eine Quasimetrik. Da sie jedoch analog zu DTW Gliedpaare dynamisch bildet, ist dies nicht der Fall: $\mathrm{d}_{\mathrm{PF}} \in \mathrm{MTR}_{\mathrm{P}}$.

Das flächenbasierte Ähnlichkeitsmaß $\mathrm{d}_{\mathrm{ABD}}$ erfüllt alle Eigenschaften einer Metrik, wenn man vom Grenzfall mit zwei eingliedrigen Trajektorien absieht, bei dem das Identitätsprinzip verletzt wäre. Auch $\mathrm{d}_{\mathrm{DISSIM}}$ ist metrisch, weil wir davon ausgehen können, dass die verwendete Wurzelfunktion zur Beschreibung der Distanz zwischen den Trajektorien nur positive Werte annimmt. Die Tatsache, dass $\mathrm{d}_{\mathrm{SPM}}$ Gewichte für die Summanden zulässt, disqualifiziert sie im Allgemeinen für die Erfüllung der metrischen Eigenschaften, denn bei bestimmten Belegungen werden sowohl Nicht-Negativität als auch das Identitätsprinzip verletzt. Selbst im speziellen Fall von geeigneten Belegungen verhindert die Minimumsfunktion der Positionsdistanz, dass wir eine Metrik erzeugen.

$\mathrm{d}_{\mathrm{AAL}}$ ist ein rotationsinvariantes Unähnlichkeitsmaß. Zwei Trajektorien, die sich durch nichts als eine Rotation unterscheiden, haben also zwangsweise die Unähnlichkeit 0 und widerlegen somit die Erfüllung des Identitätsprinzips. Das gilt auch für $\mathrm{d}_{\mathrm{SpADe}}$. Die Threshold-Distanz $\mathrm{d}_{\mathrm{TQuEST}}$ ist, wie man leicht sieht, ebenfalls keine Metrik, weil auch zwei unterschiedliche Trajektorien zu exakt gleichen Zeitpunkten die gegebene Schwelle überschreiten können. Die Road-Network-Distanzen verletzen allesamt das Identitäts-

prinzip, da es für eine Unähnlichkeit von 0 bereits ausreicht, wenn sich die Glieder der beiden Trajektorien nur in den durch die PoI beziehungsweise ToI angefragten Positionen beziehungsweise Zeitpunkten nicht unterscheiden. Es gilt also $d_{RN\_T}, d_{RN\_S}, d_{RN\_TS} \in MTR_0$. Tiakas et al. haben die metrischen Eigenschaften ihres Unähnlichkeitsmaßes d_{Graph} selbst bewiesen [TPN+09].

6.8 Komplexität der Berechnung

Klassifizierung

Die Komplexität der Berechnung ist, wie schon erwähnt, ein wesentliches Kriterium für die Auswahl eines Ähnlichkeitsmaßes. Wir betrachten insbesondere die Zeitkomplexität, also die Anzahl der benötigten Rechenschritte in Abhängigkeit der Länge der Trajektorien.

Die Klassifizierung der Ähnlichkeitsmaße findet mithilfe des Landau-Symbols $\mathcal{O}$ statt – der asymptotischen oberen Schranke.

Definition 6.11 (Zeitkomplexität der Berechnung) *Seien σ und τ zwei Trajektorien der Länge n beziehungsweise m. Ohne Beschränkung der Allgemeinheit sei dabei $n \geqslant m$. Ein Ähnlichkeitsmaß d ist in der Komplexitätsklasse* DTIME(g) *für eine Funktion g, genau dann, wenn für die Anzahl der Rechenschritte $f(n, m)$, die bei seiner Berechnung ausgeführt werden müssen, gilt:*

$$f \in \mathcal{O}(g) \Leftrightarrow \exists c > 0 \, \exists x_0 > 0 \, \forall x > x_0 : f(n, m) < c \cdot g(n)$$

Die kleinste Funktion g, die dies erfüllt, gibt die Klasse für die Komplexität des Ähnlichkeitsmaßes d an.

Maßgeblich ist nach dieser Definition die Länge der längeren Trajektorie.

Klassierung

Es gibt einige Ähnlichkeitsmaße, die die Distanz zwischen jedem Glied der ersten Trajektorie zu jedem Glied der zweiten Trajektorie berechnen und umgekehrt. Wir haben es also mit einer sehr hohen Zeitkomplexität von $\mathcal{O}(n \cdot m)$ zu tun. Das betrifft insbesondere d_{CPD}, d_{AP_mean} und $d_{Hausdorff}$ auf diskreten Trajektorien. Diese Ähnlichkeitsmaße sind also in der quadratischen Klasse, das heißt für g wird $g(n) = n^2$ gewählt. In der Übersicht notieren wir dies mit $\mathcal{O}(n^2)$. Die modifizierte Hausdorff-Distanz d_{MOHD} ist etwas effizienter zu berechnen, da die jeweils andere Trajektorie für das aktuelle Glied der

einen Trajektorie nur in dem Bereich betrachtet wird, der durch die Funktion N gegeben ist. Hat dieser Bereich durchschnittlich eine Größe von w, ist die Komplexität des Unähnlichkeitsmaß durch $\mathcal{O}(nw)$ beschreibbar. Wir stellen generell fest, dass man bei parametrierbaren Ähnlichkeitsmaßen der Klasse $\mathrm{PRM}_{\mathcal{W}}$ oft erheblichen Einfluss auf die Komplexität hat, weil so die Anzahl der Gliedpaarungen bei der Berechnung beschränkt werden kann.

Aggregate über synchrone Glieder lassen sich im Vergleich dazu recht effizient ermitteln, da sie unelastisch sind und so für jedes Glied nur eine Positionsdistanz berechnet werden muss. Die Berechnung des Aggregats selbst kann dabei vernachlässigt werden, sodass wir eine lineare Komplexität ($\mathcal{O}(n)$) erhalten [WMD+13, Abschn. 4.1]. Wir können allgemein festhalten, dass jedes unelastische Ähnlichkeitsmaß (Abschnitt 6.3) eine Zeitkomplexität von $\mathcal{O}(n)$ hat.

Einige weitere Ähnlichkeitsmaße basieren auf euklidischen Distanzen und haben daher die gleiche oder eine vergleichbare Komplexität. Dazu gehören etwa $\mathsf{d}_{\mathrm{SBD\_S}}$ und $\mathsf{d}_{\mathrm{SBD\_TS}}$. Man sollte allerdings beachten, dass die Berechnung dennoch langsamer sein wird, da die Trajektorien vorerst einem Resampling unterzogen werden – der Normalisierung.

Die Berechnung der Fréchet-Distanz, wie sie ursprünglich vorgeschlagen wurde, hat folgende Komplexität: $\mathcal{O}(nm\log(nm))$ [Bri14]. Wir können insbesondere für etwa gleich lange Trajektorien diese Abschätzung als obere Schranke vornehmen: $\mathcal{O}(nn\log(nn)) = \mathcal{O}(n^2\log(n))$, denn es gilt $\log(n^2) = 2\cdot\log(n) \in \mathcal{O}(\log(n))$. Die diskrete Fréchet-Distanz hat eine Komplexität von $\mathcal{O}(n^2)$ [EM94, Kap. 3]. Die meisten Algorithmen zur Berechnung der Fréchet-Distanz haben eine Laufzeit von $\mathcal{O}(n^2/\log n)$. Es gibt allerdings keine bekannten unteren Schranken, das heißt es ist nicht bekannt, ob es nicht bessere Algorithmen gibt. Unter Annahme der starken *Exponential Time Hypothesis*, die besagt, dass es keinen Algorithmus mit Laufzeit $\mathcal{O}((2-\delta)^{\mathsf{N}})$ für ein $\delta > 0$ zur Lösung von CNF-SAT gibt, kann allerdings auch die Fréchet-Distanz nicht in echt subquadratischer Zeit, also in $\mathcal{O}(n^{2-\delta})$ für ein $\delta > 0$, berechnet werden. Dies gilt sowohl für die kontinuierliche als auch für die diskrete Fréchet-Distanz. [Bri14]. Es ist also sehr unwahrscheinlich, dass sie effizienter berechenbar ist. Ganz ähnlich können wir annehmen, dass es keine Möglichkeit gibt, die *Edit Distance* (Abschnitt 5.3.8) effizienter als in quadratischer Zeit zu berechnen [BI14]. Das gilt somit auch für die dazugehörigen Unähnlichkeitsmaße: $\mathsf{d}_{\mathrm{EDR}}, \mathsf{d}_{\mathrm{ERP}} \in \mathcal{O}(n^2)$. Aufgrund der ähnlichen rekursiven Definition haben auch $\mathsf{d}_{\mathrm{LCSS\_1}}$ und $\mathsf{s}_{\mathrm{Swale}}$ diese Komplexität. LCSS selbst ist tatsächlich ein NP-hartes Problem [Mai78]. Weil $\mathsf{d}_{\mathrm{LCSS\_2}}$ im Vergleich zu $\mathsf{d}_{\mathrm{LCSS\_1}}$ durch die zu beachtenden Translationen offensichtlich sehr aufwendig zu berechnen ist, geben die Autoren selbst Hinweise für einen effizienten Algorithmus.

Damit erreichen sie eine Zeitkomplexität von $\mathcal{O}((n+m)^3\delta^3)$ für ein *warping window* der Größe δ [VKG02, Abschn. 3].

Die Berechnung von d_{DTW} ist ebenfalls im besten Fall quadratisch [Che05]. Wir können sie allerdings auf $\mathcal{O}(nw)$ reduzieren bei der Verwendung eines *warping windows* der Größe w [WMD+13, Abschn. 4.1]. Eine typische Größe von w ist $0.2n$ bis $0.3n$, was für die meisten Anwendungen ausreichend ist [VGD04, Kap. 6]. Mit Techniken wie Fast-search-method-for-Dynamic-Time-Warping (FTW) [SYF05] kann die Auswertung stark beschleunigt werden, indem der Suchraum eingeschränkt wird. Die Autoren von d_{AAL} stellen in [VGD04] neben ihrer Vergleichstechnik auch eine Methode vor, effizienter mit DTW umzugehen, indem sie eine untere Schranke angeben. Der Name dieser Methode ist LB_{WARP}. Es gibt zahlreiche Techniken zur Abschätzung von DTW nach unten. Tatsächlich wurde nachgewiesen, dass die amortisierten Kosten von DTW sogar linear sind [KR05]. Die Komplexität der PF-Distanz hängt wie DTW von einer Art *warping window* ab, das durch den Parameter δ bestimmt wird: $d_{PF} \in \mathcal{O}(\delta \cdot n)$.

Allgemein können wir feststellen, dass die Komplexität von elastischen Ähnlichkeitsmaßen höher ist als die von unelastischen. Interessanterweise nähern sich die amortisierten Kosten von elastischen Ähnlichkeitsmaßen mit steigender Größe des Datensatzes denen von unelastischen an. Umgekehrt nähert sich allerdings auch die Präzision bei der Bestimmung von ähnlichen Trajektorien bei elastischen Ähnlichkeitsmaßen mit steigender Größe des Datensatzes denen von unelastischen an [DTS+08, Abschn. 4.1].

d_{ABD} ist mit der Berechnung der Schnittpunkte der Trajektorien, der Entfernung der Zyklen, der Ermittlung der Regionen und ihrer anschließenden Flächenberechnung ein sehr teures Unähnlichkeitsmaß. Es wird in der Originalpublikation daher auch nur verwendet, um andere, effizientere Unähnlichkeitsmaße miteinander zu vergleichen. Zu seiner genauen Komplexität werden keine Aussagen gemacht [NB03]. Auch d_{DISSIM} ist sehr teuer, weil die Gleichungen aus Abschnitt 5.3.14 zur Berechnung der Integrale über die Beschreibung der euklidischen Distanz $D(t)$ gelöst werden müssen. Die Autoren sind sich dessen jedoch bewusst und schlagen daher zwei Abschätzungen vor, die auf der Trapezregel basieren. Eine untere Schranke für die tatsächliche Distanz wird beschrieben durch $d_{PESDISSIM}$, eine obere Schranke durch $d_{OPTDISSIM}$, sodass gilt: $d_{PESDISSIM} \leqslant d_{DISSIM} \leqslant d_{OPTDISSIM}$. Diese beiden Abschätzungen sind im Vergleich deutlich schneller zu berechnen. In einem Experiment schaffen es die Autoren so, besser als d_{LCSS} und d_{EDR} zu sein [FGT07, Abschn. 5.2].

Bei d_{SPM} werden, wie in Abschnitt 5.3.15 erläutert, drei Distanzen berechnet. Da zwei von ihnen – die euklidische und die Richtungsdistanz – lineare Komplexität haben, die dritte sogar konstante, ergibt sich auch insgesamt nur $\mathcal{O}(n)$. Die Effizienz von d_{SpADe} hängt von

dem Schwellwert für den Match von lokalen Patterns ϵ ab [CNOT07, Abschn. 5.2]. In einem Experiment ist das Unähnlichkeitsmaß effizienter als d_{DTW} und d_{EDR} [CNOT07, Abschn. 5.4].

Die Autoren von d_{TQuEST} haben dieses Ähnlichkeitsmaß entworfen, um Anfragen mit einer gegebenen Schwelle θ für die Raumdimension effizient beantworten zu können. Tatsächlich hat der Algorithmus für dessen Berechnung eine Komplexität von $\mathcal{O}(N_q \cdot N_k \cdot \log(N_p))$, wobei N_q die Größe der TCT, N_k die Anzahl der Iterationen für die Nachbarschaftssuche und N_p die Anzahl der Segmente im Merkmalsraum ist. Es gilt angeblich $N_q \ll n$ für durchschnittliche N_q und auch N_k soll sehr klein sein. N_p ist kleiner als die Summe der Längen der Trajektorien in der Datenbank [AKK+06, Kap. 5].

Bei der RN-Technik geschieht das Finden von ähnlichen Trajektorien in zwei Schritten, wie in Abschnitt 5.4.3 beschrieben. Der erste von beiden, also die Verwendung der Road-Network-Ähnlichkeit (*filtering*) hat eine Zeitkomplexität von $\mathcal{O}(n^2)$, der zweite, die Verwendung der Road-Network-Distanz (*refining*) von $\mathcal{O}(n)$ [HKL06, Kap. 4]. Wir bezeichnen die Komplexität der Road-Network-Distanz nur eingeschränkt als linear, weil sie ohne vorhergehende Anwendung der Road-Network-Ähnlichkeit nicht praktikabel ist. Die Komplexität von d_{Graph} ist zwar in Abhängigkeit der Länge der Trajektorien linear, hängt jedoch von der Kostenfunktion im Graphen ab. Ist diese nicht materialisiert, muss man deren Berechnung, zum Beispiel mit dem Dijkstra-Algorithmus, berücksichtigen. Die Grid-basierte Distanz d_{Grid} lässt sich in $\mathcal{O}(ni)$ berechnen, wobei i die durchschnittliche Anzahl der lokalen Minima – das sind die Grid-Zellen einer Grid-Trajektorie, die eine geringere Distanz zu einer gegebenen Grid-Zelle haben als ihre Nachbarn – ist. Dies ist subquadratisch [LS05, Abschn. 4.2].

In der Praxis lässt sich natürlich oft die Geschwindigkeit der Berechnung im Rahmen der theoretisch möglichen Komplexität noch erhöhen. So kann die euklidische Distanz im Vergleich zum naiven Ansatz viermal so schnell berechnet werden, indem eine Abschätzung nach unten mit der sogenannten LB_Keogh-Funktion [KWX+06] vorgenommen wird. Diese Technik ist auch auf DTW und LCSS anwendbar, jedoch nur bis zu einer zweidimensionalen Raumdimension [FGT07, Kap. 2]. Zur Implementierung sei außerdem angemerkt, dass bei DTW, LCSS, EDR, ERP und ähnlichen elastischen Ähnlichkeitsmaßen in der Regel von dynamischer Programmierung Gebrauch gemacht wird, um die Effizienz zu erhöhen [WMD+13, Kap. 2]. Eine Ausnahme stellt s_{Swale} dar, für das die Autoren eigens FTSE entwickelt haben (Abschnitt 5.3.10). Das erlaubt im Vergleich zu dynamischer Programmierung eine 7- bis 8-mal schnellere Auswertung; im Vergleich zu Techniken, die den Warping-Bereich beschränken, eine 2- bis 3-mal schnellere [MP07, Abschn. 6.1].

6.9 Samplinginvarianz

Klassifizierung

Wenn zwei Trajektorien unterschiedliche – aber durchaus jeweils für sich äquidistante – Zeitintervalle haben, ist ihre Ähnlichkeit nicht immer einfach festzustellen. Der problematische Fall tritt auf, wenn diese Trajektorien durch unterschiedliche Samplingraten entstanden sind. Veranschaulicht wird dieses Phänomen in Abbildung 6.2. Die beiden Trajektorien T und Q haben augenscheinlich eine ähnliche Form und ein ähnliches Akkumulat, obwohl T die Länge 5 und Q die Länge 33 hat. Ihre mit einer Technik wie DTW berechnete Unähnlichkeit wird in einem solchen Fall zu groß sein, da jede Position von T einzeln in das Ergebnis einfließt.

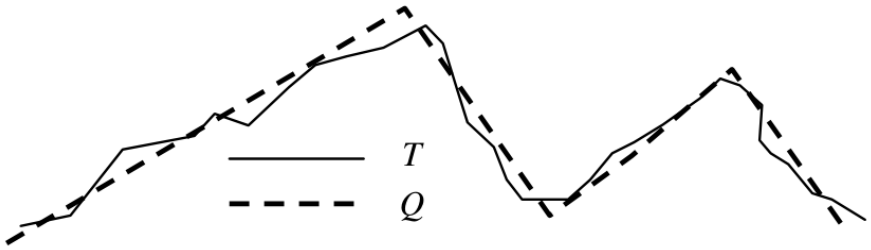


Abbildung 6.2: Zwei Trajektorien mit unterschiedlichen Samplingraten [FGT07].

Definition 6.12 (Samplinginvarianz) *Seien* σ *und* τ *zwei für ein Ähnlichkeitsmaß* d *geeignete Trajektorien. Wir definieren die Samplinginvarianz (**SI**) wie folgt:*

- ***I**: Man kann ein Resampling von* τ *durchführen und für alle möglichen Resamplings* τ' *von* τ *gilt:* $\mathrm{d}(\sigma, \tau') = \mathrm{d}(\sigma, \tau)$*. Wir sagen:* d *ist **samplinginvariant**.*
- ***0**:* d *ist nicht samplinginvariant.*

Ein Unähnlichkeitsmaß ist demnach samplinginvariant, wenn die Samplingstruktur der interpolierten Linien einer Trajektorie keinen Einfluss auf ihr Ergebnis nimmt. Wir schließen mit der Definition von $\mathrm{SI_I}$ Ähnlichkeitsmaße, die identische Zeitstempel benötigen, aus, indem nur eine der beiden Trajektorien normalisiert wird: $\mathrm{ZS_I} \subseteq \mathrm{SI_0}$.

Der Zweck eines samplinginvarianten Ähnlichkeitsmaßes in der Praxis ist seine Verwendbarkeit auf Trajektorien mit unterschiedlichen Samplingraten, ohne dass ein gesondertes Resampling nötig ist.

Klassierung

Weder Aggregate über synchrone Glieder noch die euklidische Distanz noch die üblichen elastischen Ähnlichkeitsmaße noch deren Verwandte sind samplinginvariant: $d_{AI}, d_{E}, d_{PF}, d_{DTW}, d_{LCSS\_1}, d_{LCSS\_2}, d_{LCSS\_SM\_1}, d_{LCSS\_SM\_2}, d_{EDR}, d_{ERP}, d_{Swale} \in SI_0$.

Hingegen sind Ähnlichkeitsmaße, denen ein kontinuierliches Modell zugrunde liegt – die Area-Based-Distance eingeschlossen –, samplinginvariant: $d_{Hausdorff}, d_{MOHD}, d_{Fréchet}, d_{ABD}, d_{DISSIM} \in SI_I$. Die Closest-Pair-Distance und Aggregate über Punkt-Trajektorien-Distanzen und die Grid-basierte Distanz sind samplinginvariant, weil sich durch ein Resampling die minimale Distanz zum nächsten Glied nicht ändern kann: $d_{CPD}, d_{AP}, d_{Grid} \in SI_I$. Die Shape-Based-Distance ist deswegen samplinginvariant, weil zur ihr eine zeitliche beziehungsweise räumliche Normalisierung gehört. Die normalisierte Trajektorie eines Resamplings ein und derselben Trajektorie ist stets identisch. Daher gilt: $d_{SBD\_TS}, d_{SBD\_S} \in SI_I$.

Bei Ähnlichkeitsmaßen der Klasse MR_M ist die Fragestellung nach der Samplinginvarianz komplexer. d_{SpADe} zum Beispiel berechnet seinen Wert anhand von lokalen Patterns, sodass unterschiedliche Zeitstempel, die fehlende Samplinginvarianz verursachen, nicht vorkommen. Aus $d_{AI\_RMS} \in SI_0$ folgt auch $d_{SPM} \in SI_0$ für Sequence-Pattern-Mining. Ebenso folgt aus $d_{DTW} \in SI_0$ auch $d_{AAL} \in SI_0$ für AAL-Warping. Darüber hinaus gibt es Räume, bei denen Resampling keine sinnvolle Operation ist, so etwa bei den Road-Network-Distanzen. Dann ist auch eine Klassierung bezüglich der Samplinginvarianz nicht sinnvoll. Anders sieht es bei d_{TQuEST} aus. Obwohl hier wie in Abschnitt 6.1 erwähnt ein Merkmalsraum verwendet wird, können wir die Veränderung des Wertes durch Resampling sicher ausschließen, denn die Zeitstempel der TCT werden sich dadurch nicht ändern. Daher gilt: $d_{TQuEST} \in SI_I$.[4]

6.10 Empfindlichkeit auf Ausreißer

Klassifizierung

Unter einem **Ausreißer** verstehen wir ein Glied einer Trajektorie, dessen Position so stark von den anderen Positionen der Trajektorie abweicht, dass die Authentizität und Korrektheit des Wertes in Frage gestellt werden muss. In der Praxis sind solche Ausreißer aufgrund von fehlerhaften Messungen oder ähnlichen Ursachen häufig anzutreffen.

[4] Wir gehen an dieser Stelle von linearer Interpolation aus, wenn der Schwellenwert nicht zufällig den Positionen der Glieder entspricht.

Deswegen ist es sinnvoll, sich Gedanken dazu zu machen, wie ein Ähnlichkeitsmaß mit Ausreißern umgeht.

Definition 6.13 (Empfindlichkeit auf Ausreißer) *Sei* d *ein Ähnlichkeitsmaß auf Trajektorien. Seine Empfindlichkeit auf Ausreißer (**AR**) kategorisieren wir wie folgt:*

- ***I**: Ein Ausreißer verfälscht das Ergebnis von* d *immer signifikant.*
- ***M**: Ein Ausreißer kann das Ergebnis von* d *signifikant verfälschen, tut dies aber nicht zwingend.*
- ***0**: Ausreißer werden stets sinnvoll behandelt.*

Die Klassen sind paarweise disjunkt.

Mit der Empfindlichkeit auf Ausreißer hängt auch die Empfindlichkeit auf **Rauschen**, das ist eine über die gesamte Trajektorie verteilte Störung, die dessen Positionen verfälscht, zusammen. Wie letztere das Ergebnis eines Ähnlichkeitsmaßes beeinflusst, hängt allerdings maßgeblich von der Klassenzugehörigkeit der Akkumulation (AK) ab, denn gegebenenfalls werden Fehler – so bei der euklidischen Distanz – akkumuliert [CNOT07, Kap. 2]. Je nach Art des Rauschens kann dies jedoch sogar zur Auslöschung des Fehlers führen, so etwa bei der euklidischen Distanz und weißem gaußschem Rauschen [AFS93, Abschn. 3.1].

Klassierung

$\mathsf{d}_{\mathrm{CPD}}$ gehört zur Klasse AK_{E} und hat die Eigenschaft, einen Ausreißer einer Trajektorie ignorieren zu können, wenn dessen Position weit von der anderen Trajektorie entfernt liegt. Liegt er hingegen näher als andere Positionen an ihr, verfälscht er das Ergebnis maßgeblich. Daher gilt: $\mathsf{d}_{\mathrm{CPD}} \in \mathrm{AR}_{\mathsf{M}}$. Auf die gleiche Weise können wir diese Zuordnung für $\mathsf{d}_{\mathrm{AI\_min}}$ und $\mathsf{d}_{\mathrm{AI\_max}}$ vornehmen. Es gilt jedoch keineswegs $\mathrm{AK}_{\mathsf{E}} \subseteq \mathrm{AR}_{\mathsf{M}}$, wie wir an der Hausdorff-Distanz erkennen. Ein Ausreißer in Richtung der zweiten Trajektorie verfälscht das Ergebnis, weil die einseite Hausdorff-Quasimetrik (h) ausgehend von der zweiten Trajektorie deutlich zu klein wäre; ein Ausreißer in die andere Richtung verfälscht das Ergebnis, weil selbige ausgehend von der ersten Trajektorie deutlich zu groß wäre. Also gilt: $\mathsf{d}_{\mathrm{Hausdorff}} \in \mathrm{AR}_{\mathrm{I}}$. Genauso kann man bei der Fréchet-Distanz argumentieren, denn jeder Ausreißer verändert die Länge der nötigen „Hundeleine“ entscheidend. Die modifizierte Hausdorff-Distanz löst dieses Problem durch den Parameter α, der es erlaubt, dass solche einzelnen Ausreißer nicht in das Ergebnis einfließen, weil das Supremum durch $\mathrm{ord}^{\alpha}_{s\in\sigma}$ ersetzt wird. Daher gilt: $\mathsf{d}_{\mathrm{MOHD}} \in \mathrm{AR}_0$.

Mit $\mathsf{d}_{\mathrm{CPD}}$ haben wir ein Beispiel der Schnittmenge $\mathrm{AK}_{\mathsf{E}} \cap \mathrm{AR}_{\mathsf{M}}$, mit $\mathsf{d}_{\mathrm{Hausdorff}}$ eines der Schnittmenge $\mathrm{AK}_{\mathsf{E}} \cap \mathrm{AR}_{\mathrm{I}}$. Das Median-Aggregat über synchrone Glieder $\mathsf{d}_{\mathrm{AI\_median}}$ ist neben der modifizierten Hausdorff-Distanz ein Beispiel dafür, dass auch der Schnitt $\mathrm{AK}_{\mathsf{E}} \cap \mathrm{AR}_0$ nicht leer ist.

Bei den euklidischen Distanzen, die der Klasse AK_{A} angehören, fließt der Fehler immer ein. Es gibt keinen Fall, in dem ein Ausreißer das Ergebnis nicht verfälschen würde: $\mathsf{d}_{\mathrm{AI\_sum}}, \mathsf{d}_{\mathrm{AI\_mean}}, \mathsf{d}_{\mathrm{AI\_RMS}}, \mathsf{d}_{\mathrm{E}} \in \mathrm{AR}_{\mathrm{I}}$. Auch bei $\mathsf{d}_{\mathrm{DTW}}$ ist dies nicht anders.

Während euklidische Distanzen und DTW also nicht mit Ausreißern umgehen können, vermag LCSS dies sehr wohl [CNOT07]. Das liegt einerseits daran, dass die entsprechenden Glieder schlicht nicht gepaart werden. Es besteht also sehr wohl Zusammenhang zu den Klassen der Akkumulation (AK) und Elastizität (EL). Andererseits hängt die Empfindlichkeit aber hauptsächlich von der Art ab, wie aus den Positionsdistanzen ein Ergebnis berechnet wird, präziser: von der Maßdimension (MD). LCSS kann deswegen mit Ausreißern umgehen, weil selbst unüblich große Positionsdistanzen auf einen nicht unüblichen Wert von 1 quantisiert werden. Das gilt ebenfalls für deren verwandte Techniken EDR und Swale: $\mathsf{d}_{\mathrm{LCSS\_1}}, \mathsf{d}_{\mathrm{LCSS\_2}}, \mathsf{d}_{\mathrm{LCSS\_SM\_1}}, \mathsf{d}_{\mathrm{LCSS\_SM\_2}}, \mathsf{d}_{\mathrm{EDR}}, \mathsf{s}_{\mathrm{Swale}} \in \mathrm{AR}_0$. Wir beobachten: $\mathrm{MD}_0 \subseteq \mathrm{AR}_0$. Dagegen machen Ausreißer bei ERP durch den Einfluss tatsächlicher Punktdistanzen wieder einen Unterschied: $\mathsf{s}_{\mathrm{ERP}} \in \mathrm{AR}_{\mathrm{I}}$.

Die PF-Distanz ist ein interessanter Fall, denn sie ist – wie wir aus Abschnitt 6.7 wissen – nicht symmetrisch. Das hat zur Folge, dass Ausreißer der zweiten Trajektorie bei zu großer Distanz sinnvoll behandelt werden könnten, da sie nicht mit Gliedern der ersten Trajektorie gepaart werden müssen. Für Ausreißer in Richtung der ersten Trajektorie gilt dies nicht, weil sie die Distanz verkleinern. Außerdem verfälscht jeder Ausreißer der ersten Trajektorie, egal in welche Richtung – das Ergebnis. Wir müssen also konsequenterweise die folgende Zuordnung vornehmen: $\mathsf{d}_{\mathrm{PF}} \in \mathrm{AR}_{\mathrm{I}}$.

Die meisten anderen aufgeführten Ähnlichkeitsmaße reagieren empfindlich auf Ausreißer. AAL-Warping zum Beispiel ist genauso empfindlich wie das bei dieser Technik verwendete DTW selbst [Che05], die Fläche bei den flächenbasierten Unähnlichkeitsmaßen $\mathsf{d}_{\mathrm{ABD}}$ und $\mathsf{d}_{\mathrm{DISSIM}}$ wird größer. Bei $\mathsf{d}_{\mathrm{SBD\_TS}}$ kann es allerdings tatsächlich vorkommen, dass Ausreißer keine Rolle spielen, nämlich genau dann, wenn die zeitliche Normalisierung mit so großen Zeitintervallen durchgeführt wird, dass die kritischen Positionen nicht in die Interpolation einfließen. SpADe ist in der Lage, mit Ausreißern umzugehen [CNOT07, Abschn. 3.2]. Die Grid-basierte Distanz $\mathsf{d}_{\mathrm{Grid}}$ ist genau empfindlich gegenüber Ausreißern wie die zugrunde liegende Distanz $\mathsf{d}_{\mathrm{AP\_mean}}$. Bei den Netzwerk- und Graph-basierten Distanzen ist eine Klassierung nicht sinnvoll, weil Ausreißer bei dieser Art der Trajektorien, also dem Raum mit diskreten Positionen, keine Entsprechung haben.

Zuletzt sei erwähnt, dass es einen wesentlichen Zusammenhang zwischen den metrischen Eigenschaften (MTR) eines Unähnlichkeitsmaßes und seinem Vermögen, mit Ausreißern und Rauschen umzugehen, gibt. Damit einher geht nämlich die Erfüllung der Dreiecksungleichung:

> „Distance functions that are robust to outliers or to extremely noisy data will typically violate the triangle inequality. [...] This violation of the triangle inequality is not an artifact of a poor choice of features. It is inherent in the idea of robust matching, which allows one portion of an object to be matched to one image, and a different portion to match a different image. These objects cannot be mapped into a metric feature space without large distortions in the distances between them.“ [JWG00, Abschn. 2.1]

Wenn man also ein robustes Ähnlichkeitsmaß braucht, das Ausreißer sinnvoll behandelt, muss man zwangsweise seine Verletzung metrischer Eigenschaften in Kauf nehmen. Nach Ansicht von Jacobs et al. kann man sogar Probleme als inhärent nichtmetrisch ansehen [JWG00, Abschn. 2.2].

6.11 Transformationsinvarianz

Es ist häufig erwünscht, dass ein Unähnlichkeitsmaß für Trajektorien oder Zeitreihen invariant unter geometrischen Transformationen ist. Abbildung 6.3 verdeutlicht dies. Die beiden Zeitreihen A und B sind sich intuitiv ähnlicher als A und C, weil B entweder lediglich eine Translation (a) beziehungsweise Skalierung (b) zu A erfahren hat. Trotzdem wird zum Beispiel die mittels DTW berechnete Distanz zwischen A und B größer sein als zwischen A und C.

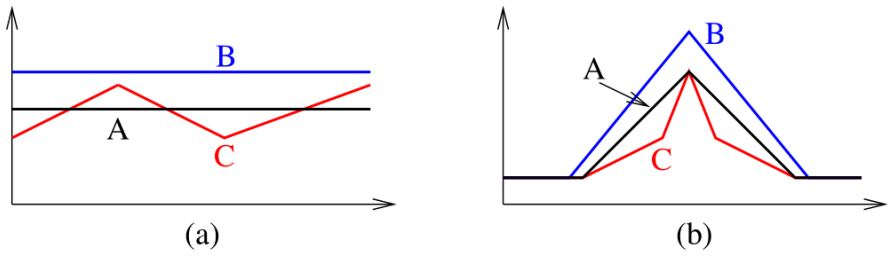


Abbildung 6.3: Translation und Skalierung von Zeitreihen [CNOT07].

Klassifizierung

In Abschnitt 4.1.3 haben wir spezifiziert, was mit den Transformationen *Translation*, *Rotation* und *Skalierung* gemeint ist und wann ein Ähnlichkeitsmaß invariant unter solchen Transformation ist. Das führt uns rasch zu der folgenden Klassifikation:

Definition 6.14 (Invarianz unter Transformationen) *Sei* d *ein Ähnlichkeitsmaß auf Trajektorien. Bezüglich seiner Transformationsinvarianz (**TI**) klassifizieren wir wie folgt:*

- ***L**:* d *ist translationsinvariant gemäß Definition 4.8.*
- ***R**:* d *ist rotationsinvariant gemäß Definition 4.9.*
- ***S**:* d *ist skalierungsinvariant gemäß Definition 4.10.*
- ***0**:* d *ist weder translationsinvariant, noch rotationsinvariant oder skalierungsinvariant.*

Die Klasse TI_0 schließt alle anderen Klassen aus: $\mathrm{TI}_0 \cap (\mathrm{TI}_\mathsf{L} \cup \mathrm{TI}_\mathsf{R} \cup \mathrm{TI}_\mathsf{S}) = \emptyset$. Die Klassen TI_L, TI_R und TI_S sind offenkundig nicht paarweise disjunkt.

Klassierung

Eine Klassifizierung zur Invarianz unter Transformation ist nur sinnvoll, wenn der Raum der Trajektorien solche Transformationen zulässt. Das schließt insbesondere die Road-Network-Distanzen, die Graph-basierte Distanz und Distanzen auf semantischen Trajektorien aus.

Wenige der vorgestellten Unähnlichkeitsmaße sind translationsinvariant, rotationsinvariant oder skalierungsinvariant im Sinne der oben definierten Klassifikation, inbesondere nicht die „einfacheren“: $\mathrm{d}_{\mathrm{CPD}}, \mathrm{d}_{\mathrm{E}}, \mathrm{d}_{\mathrm{AP\_mean}}, \mathrm{d}_{\mathrm{Hausdorff}}, \mathrm{d}_{\mathrm{Fréchet}} \in \mathrm{TI}_0$. Bei einigen Techniken gehört eine optimale Translation zur Minimierung der Unähnlichkeit jedoch dazu, sodass sie translationsinvariant sind: $\mathrm{d}_{\mathrm{LCSS\_2}}, \mathrm{d}_{\mathrm{LCSS\_SM\_2}} \mathrm{d}_{\mathrm{ABD}} \in \mathrm{TI}_\mathsf{T}$. Es muss allerdings nicht immer eine optimale Transformation im Raum der Trajektorien stattfinden, um ein Ähnlichkeitsmaß invariant unter einer solchen Transformation zu machen. Bei der AAL-Warping-Distanz wird ein Merkmalsraum verwendet, der die gewünschten Eigenschaften hat, sodass die dort berechnete Ähnlichkeit sich unter Translation, Rotation und Skalierung nicht ändert. Rotationen im Raum der Trajektorien führen zu einer Translation im AAL-Raum. Skalierungsinvarianz wird erreicht, indem die Länge jeder Strecke schlicht durch das Akkumulat der Trajektorie geteilt wird, zu der diese Strecke gehört [VGD04, Kap. 3.1]. $\mathrm{d}_{\mathrm{AAL}} \in \mathrm{TI}_\mathsf{T} \cap \mathrm{TI}_\mathsf{R} \cap \mathrm{TI}_\mathsf{S}$. In Abschnitt 6.7 haben wir festgestellt, dass $\mathrm{d}_{\mathrm{AAL}}$ nicht metrisch ist, weil seine Rotationsinvarianz zwangsweise das Identitätsprinzip ver-

letzt. Weil dies auch für Translation und Skalierung gilt, können wir allgemein festhalten, dass solche Ähnlichkeitsmaße niemals metrisch sind: $(\mathrm{TI_T} \cup \mathrm{TI_R} \cup \mathrm{TI_S}) \cap \mathrm{MTR_M} = \emptyset$.

Das Unähnlichkeitsmaß $\mathrm{d_{SpADe}}$ wird mithilfe von aus Trajektorien erzeugten lokalen Patterns berechnet. Es scheint zwar nicht für alle eingegebenen Trajektorien translationsinvariant, jedoch nicht so empfindlich bei Translation zu sein. Translation entspricht hier der in der Publikation „amplitude shifting“ genannten Verschiebung in der Raumdimension [CNOT07].

6.12 Eignung für inkrementelle Berechnung

Unter **inkrementeller Berechnung**[5] eines Ähnlichkeitsmaßes verstehen wir seine wiederholte aktualisierende Berechnung, während die Trajektorien noch weitere Verlängerungen durch neue Glieder erfahren. Ein solches Verfahren ist für eine Vielzahl an Anwendungen nützlich, wenn man Interesse an den Positionen in Echtzeit hat, zum Beispiel bei der Überwachung von Flugobjekten und Erkennung von Patterns [CNOT07]. Werden Trajektorien in Echtzeit erhoben, möchte man die Daten dann in der Regel komprimieren. Dies geschieht zum Beispiel dadurch, dass nur neue Glieder hinzugefügt werden, wenn sich die Position des bewegten Objektes hinreichend geändert hat. Um eine solche Veränderung festzustellen, muss die Ähnlichkeit der Trajektorien berechnet werden. Üblicherweise geschieht dies mit einer Art der euklidischen Distanz [MR04, LRH11].

Wir verstehen die Eignung zur inkrementellen Berechnung als eigenes Charakteristikum eines Ähnlichkeitsmaßes. Anhand der Komplexität seiner laufenden Berechnung stellen wir sie fest, weil dies die interessante Eigenschaft für Echtzeit-Anwendungen ist. Eine Definition könnte in etwa so aussehen:

Definition 6.15 (Eignung für inkrementelle Berechnung) *Sei* d *ein Ähnlichkeitsmaß auf Trajektorien. Die Eignung für seine inkrementelle Berechnung (**IN**) kategorisieren wir wie folgt:*

- ***E**: Die Berechnung von* d *auf zwei Trajektorien ist möglich, obwohl noch nicht alle Glieder existieren und erfordert nur annähernd konstanten rechnerischen Aufwand und Speicher in Abhängigkeit der Länge der Trajektorien.*
- ***0**: Eine solche Berechnung ist nicht möglich oder erfordert höheren rechnerischen Aufwand oder mehr Speicher.*

In der Praxis hängt die Eignung von der konkreten Implementierung ab. Man kann zum Beispiel die Zeitkomplexität verringern, wenn man höheren Speicherbedarf in Kauf

[5] Der Begriff „On-line-Berechnung“ ist eng damit verwandt.

nimmt. Daher wäre eine schlichte und endgültige Zuordnung von Ähnlichkeitsmaßen zu diesen Klassen irreführend und anfechtbar. Allerdings gibt es Ähnlichkeitsmaße, bei denen eine einfache und effiziente inkrementelle Berechnung auf der Hand liegt.

Die Aggregate über synchrone Glieder lassen sich beispielsweise problemlos inkrementell berechnen, wie man ihrer Definition leicht ansieht. Tatsächlich gilt dies für alle vorgestellten Ähnlichkeitsmaße, die identische Zeitstempel voraussetzen: $\mathsf{d}_{\mathrm{AI}}, \mathsf{d}_{\mathrm{E}}, \mathsf{d}_{\mathrm{PF}}, \mathsf{d}_{\mathrm{SPM}}$, $\mathsf{d}_{\mathrm{RN\_T}}, \mathsf{d}_{\mathrm{RN\_S}}, \mathsf{d}_{\mathrm{RN\_TS}}, \mathsf{d}_{\mathrm{Graph}} \in \mathrm{ZS}_{\mathrm{I}} \subseteq \mathrm{IN}_{\mathrm{E}}$. Bei der Road-Network-Technik muss zwar über die Menge an PoI beziehungsweise ToI iteriert werden, nicht jedoch über die Trajektorien. Die Graph-basierte-Distanz verhält sich prinzipiell wie eine euklidische, allerdings unter der Prämisse, dass sich die Kostenfunktion des Graphen effizient ausrechnen lässt. Ist dies nicht der Fall, weil zum Beispiel bei der inkrementellen Berechnung neue Knoten hinzukommen, ist die Eignung hinfällig. Die Berechnungskosten der PF-Distanz steigen zwar mit der Länge der Trajektorien, weil das Intervall, aus dem j ausgewählt wird, größer wird, aber nur in geringem Maße. Insbesondere weil das Unähnlichkeitsmaß eigens für die inkrementelle Berechnung entworfen wurde, ordnen wir es dieser Klasse zu:

> „Our main aim is to avoid the classical two-step clustering (data collection and off-line processing)“ [PF06, Kap. 1]

Im Gegensatz zur PF-Distanz unterstützen viele elastische Ähnlichkeitsmaße eine inkrementelle Berechnung jedoch nicht per se, weil sie eine hohe Zeitkomplexität haben und auf die Vollständigkeit der Trajektorien angewiesen sind [MT09, Abschn. 2.6]. Dazu gehören DTW und die auf LCSS basierenden Ähnlichkeitsmaße und deren Verwandte. Man würde also wie folgt klassieren: $\mathsf{d}_{\mathrm{DTW}}, \mathsf{d}_{\mathrm{LCSS\_1}}, \mathsf{d}_{\mathrm{LCSS\_2}}, \mathsf{d}_{\mathrm{LCSS\_SM\_1}}, \mathsf{d}_{\mathrm{LCSS\_SM\_2}}, \mathsf{d}_{\mathrm{EDR}}, \mathsf{d}_{\mathrm{ERP}}$, $\mathsf{s}_{\mathrm{Swale}} \in \mathrm{IN}_0$. AAl-Warping ist deswegen ungeeignet, weil es sich auf DTW bezieht. Ebenso müsste bei der Closest-Pair-Distance oder der Punkt-Trajektorien-Distanz für jedes neue Glied auf die gesamten bisherigen Trajektorien zurückgegriffen werden. Sie beziehungsweise ihre Aggregate eignen sich also auch nicht auf intuitive Weise: $\mathsf{d}_{\mathrm{AAL}}, \mathsf{d}_{\mathrm{CPD}}, \mathsf{d}_{\mathrm{AP}} \in \mathrm{IN}_0$. Die flächenbasierten Unähnlichkeitsmaße $\mathsf{d}_{\mathrm{ABD}}$ und $\mathsf{d}_{\mathrm{DISSIM}}$ sind ebenfalls beide sehr teuer, also prinzipiell ungeeignet für die inkrementelle Berechnung.

Wie oben erwähnt, sind Zuordnungen aber nicht absolut. Insbesondere für DTW gibt es tatsächlich Algorithmen, die bei Durchsetzung eines *warping windows* und entsprechendem Speicherbedarf eine zeitlich effiziente Berechnung durchführen [XX15].

$\mathsf{d}_{\mathrm{SpADe}}$ eignet sich nach Angaben der Autoren für eine inkrementelle Berechnung [CNOT07, Kap. 1]. Die Threshold-Distanz $\mathsf{d}_{\mathrm{TQuEST}}$ beruft sich in seiner Definition auf das Minimum der Distanzen von TCT. Dies bedeutet an sich linear mit der Länge der Trajektorien steigende Kosten. Weil aber die Distanz zu einem der TCT der jeweils anderen Trajektorie

nicht kleiner werden kann, wenn die Zeitstempel monoton wachsen, muss für jedes neue Glied nur ein neuer Wert berechnet und mit dem aktuellen Minimum verglichen werden. Hier ist also eine inkrementelle Berechnung trotzdem effizient durchführbar.

6.13 Notwendigkeit der Vorverarbeitung

Diese Klassifikation (**PRE**) hängt stark mit der Komplexität zusammen. Manche Techniken erfordern nämlich eine Vorverarbeitung (engl. *preprocessing*) der Trajektorien, die teilweise rechenintensiver ist als die Berechnung der Ähnlichkeit selbst. Dies wollen wir nicht außer Acht lassen.

Man kann also Kategorien für Ähnlichkeitsmaße erstellen, die sie danach beurteilen, ob eine solche Vorverarbeitung notwendig ist oder nicht oder mit welcher Komplexität sie verbunden ist. Wir regen diese Kategorisierung jedoch nur an und formulieren sie nicht explizit. Das hat den Grund, dass sie in starkem Zusammenhang mit anderen Klassifikationen steht: einerseits offensichtlich mit der Komplexität (KMP), weil diese dadurch implizit erhöht wird. Manchmal ist die Vorverarbeitung sogar gewissermaßen Teil der Technik, so wie bei der Shape-Based-Distance. In diesem Fall fließt die Komplexität in die Klassierung mit ein, andernfalls nicht. Eine Kategorisierung kann hier also irreführend sein. Andererseits mit den Anforderungen an Zeitstempel und Länge (ZS, ZI, LN), weil die Vorverarbeitung oft dazu dient, diese Anforderungen zu erfüllen. Möchte man ein Ähnlichkeitsmaß verwenden, das solche Anforderungen – zum Beispiel identische Zeitstempel – stellt, und arbeitet mit Daten, die sie nicht erfüllen, kann man dies durch Vorverarbeitung – im Beispiel Resampling – umgehen. Dieses Beispiel lässt sich allerdings eher als Charakteristikum der Daten im Zusammenhang mit einem Ähnlichkeitsmaß sehen, nicht als Charakteristikum des Ähnlichkeitsmaßes selbst. Mit der Klassifikation PRE ist hingegen die abstraktere Notwendigkeit jeglicher Vorverarbeitung gemeint. Da sie sehr individuell sein kann, können wir keine Klassifikation konkretisieren. Es sei darüber hinaus der Zusammenhang mit der Klassifikation AR angemerkt, denn oft lassen sich Ausreißer durch Vorverarbeitung beseitigen [ZZ11, Abschn. 1.8].

6.14 Eignung für Subtrajektorien

Diese Klassifikation (**SUB**) formulieren wir aus ähnlichen Gründen ebenfalls nicht explizit, deuten sie aber an, weil sie interessant ist. Sie befasst sich damit, ob ein Ähnlichkeitsmaß sich dafür eignet, Ähnlichkeiten zwischen zwei Trajektorien unterschiedlicher

Länge zu finden. Damit ist gemeint, dass die erste Trajektorie eine Subtrajektorie hat, die ähnlich oder sogar identisch zu der zweiten ist.

Die Klassifikation hängt offensichtlich mit der Klassifikation der Anforderung and die Länge (LN) zusammen, aber auch mit Akkumulation (AK) und Elastizität (EL), weil beispielsweise bei Ähnlichkeitsmaßen der Klasse AK_{E} wie $\mathsf{d}_{\mathrm{CPD}}$ das Ergebnis oft unabhängig von solchen Subtrajektorien ist. Hingegen ist es bei euklidischen Distanzen, die zur Klasse $\mathrm{AK}_{\mathsf{A}} \cap \mathrm{LN}_{\mathsf{L}}$ gehören, gar nicht erst möglich, solche Ähnlichkeiten zu erkennen. Aber auch elastische, nicht überelastische Ähnlichkeitsmaße wie $\mathsf{d}_{\mathrm{DTW}}$ eignen sich nicht für die Erkennung von ähnlichen Subtrajektorien, weil die Unähnlichkeit zu groß wird, wenn alle Glieder der längeren Trajektorie gepaart werden müssen, obwohl die kürzere Trajektorie genau mit einer Subtrajektorie der längeren übereinstimmt. Sequence-Pattern-Mining wurde genau mit dem Ziel entworfen, für die Erkennung der Ähnlichkeit von Subtrajektorien geeignet zu sein. Möglich wird dies durch den Vergleich von einzelnen Segmenten (Abschnitt 5.5.1). Weil manche Techniken wie SpADe oder der Graph-basierte Ansatz explizit Segmentierung der Trajektorien beziehungsweise Zerlegung in Subtrajektorien vornehmen, die nicht Teil des Ähnlichkeitsmaßes selbst ist, ist eine simple Zuordnung zu Klassen problematisch.

7 Zusammenfassung und Diskussion

Mit diesem Kapitel wird die Arbeit abgeschlossen. Nach einer Zusammenfassung des Inhalts und der Übersicht über die Ergebnisse reflektiert eine abschließende Diskussion ihre Bedeutung.

7.1 Zusammenfassung

Spätestens mit den ersten drei Kapiteln dieser Arbeit wird deutlich, wie vielfältig sowohl die Anwendungen für den Vergleich von Pfaden von bewegten Objekten als auch die existierenden dafür verwendeten Techniken sind. Deshalb und trotz der Relevanz der Thematik ist es mühsam, diese Vergleichstechniken für Trajektorien gegeneinander abzuwägen und neue Ansätze einzuordnen (Kapitel 3).

Diese Arbeit führt eine einheitliche Terminologie ein (Kapitel 4) und stellt eine umfangreiche und kohärente Auswahl von Ähnlichkeitsmaßen vor, die es in dieser Form in der Literatur nach unserem besten Wissen nicht gibt (Kapitel 5). In sinnhafter Ordnung und Didaktik werden für den Trajektorienvergleich relevante Ansätze dargestellt. Einige davon sind im Fachgebiet wohlbekannt, so etwa die euklidischen Distanzen, die Frèchet- und Hausdorff-Distanz, DTW, LCSS, EDR und ERP; andere sind es weniger, stellen aber wie die modifizierte Hausdorff-Distanz, die PF-Distanz oder das Sequence-Weighted-Alignment-Model interessante Abstraktionen der bestehenden Techniken dar oder erfüllen wie AAL-Warping oder die Spatial-Assembling-Distance spezielle Anforderungen, die „klassische“ Techniken nicht erfüllen. Wieder andere sind besonders akkurat (Area-Based-Distance) oder besonders effizient (Similarity-search-based-on-Threshold-Queries). Es gibt sogar Techniken, die den Trajektorienbegriff in Frage stellen und ihn manipulieren (Graph-basierter Ansatz) oder ihn mit Semantik versehen (Maxmial-Semantic-Trajectory-Pattern).

Die in Kapitel 6 eingeführte Klassifizierung von Ähnlichkeitsmaßen auf Trajektorien liefert eine Systematik für und so einen Überblick über die Vielzahl der einzelnen Methoden trotz ihrer Diversität. Dies geschieht durch die Festlegung von Klassifikationen, sodass

sich die Ähnlichkeitsmaße anhand ihrer Charakteristika verschiedenen Klassen zuordnen lassen. Das so geschaffene Klassensystem für Ähnlichkeitsmaße auf Trajektorien erlaubt den präzisen und objektiven Vergleich der zugrunde liegenden Techniken. Die Klassifikationen umfassen unter anderem Anforderungen an die Trajektorien (ZS, ZI, LN) (Definitionen 6.1, 6.2 und 6.3), die Art der Berechnung (AK, EL, MD) (Definitionen 6.4, 6.5 und 6.8) oder deren Komplexität (KMP) (Definition 6.11), Zusammenhänge zwischen Trajektorien und Ergebnis (LE, ZE, AR) (Definitionen 6.6, 6.7 und 6.13) oder Eigenschaften des Ähnlichkeitsmaßes als Funktion (PRM, MTR, TI) (Definitionen 6.9, 6.10 und 6.14). Über die Definition der Klassifikationen hinaus wird ein Großteil der in Kapitel 5 aufgeführten Ähnlichkeitsmaße den erstellten Klassen in Verbindung mit einer Erläuterung zugeordnet. So wird insbesondere deren Sinnhaftigkeit und Anwendbarkeit verdeutlicht. Besagtes Klassensystem erfüllt somit den im ersten Kapitel vorgestellten Bedarf, Vergleichstechniken für Trajektorien gegeneinander abwägen zu können, und erleichtert so die Auswahl einer solchen für eine konkrete Anwendung.

7.2 Übersicht

Eine Übersicht über die im Rahmen dieser Arbeit erzielten Ergebnisse – sowohl über Ähnlichkeitsmaße auf Trajektorien als auch über die Klassifikationen und den Zusammenhang zwischen beiden – lässt sich am besten in Form von Tabellen darstellen. In der Tabelle 7.1 findet sich eine chronologische Übersicht über die in Kapitel 5 vorgestellten Techniken für den Trajektorienvergleich und die dazugehörigen Ähnlichkeitsmaße, zusammen mit Informationen zu deren Autoren und der zugrunde liegenden oder relevanten Publikationen. Die Tabellen 7.2 und 7.3 stellen eine Übersicht über Ähnlichkeitsmaße und deren Charakteristika bezüglich der wichtigsten in Kapitel 6 definierten Klassifikationen dar. Ist in einem Feld „-“ eingetragen, ist die Zuordnung zu einer Klasse für das Ähnlichkeitsmaß dieser Zeile nicht generell entscheidbar oder nicht sinnvoll. Eine Klammerung mit „(...)“ bedeutet eine eingeschränkte Zuweisung, die im Abschnitt der jeweiligen Klassifikation erläutert ist.

7.3 Diskussion

Das im Rahmen der vorliegenden Arbeit entstandene Klassensystem schließt die Lücke der fehlenden objektiven Kriterien für den Vergleich von Vergleichstechniken auf Trajektorien. Zusammen mit der vorhergehenden Darstellung von Ansätzen leistet die Arbeit einen

bedeutsamen Beitrag für das Forschungsgebiet, indem sie einen umfassenden Überblick über und eine Einordnung von Methoden für den Trajektorienvergleich erlaubt.

Weder die Darstellung der Techniken noch das vorgeschlagene Klassensystem sollten jedoch als vollständig und absolut angesehen werden. Es mag in Zukunft neue Ansätze geben, die sich tatsächlich als so qualitätvoll erweisen, dass andere Ansätze obsolet werden. Gewiss sind auch andere sinnhafte Zusammenstellungen von Vergleichstechniken denkbar. Auch mag es so spezielle Anwendungsbereiche geben, dass ihnen die Darstellung der möglichen Methoden in dieser Arbeit nicht gerecht wird. Das Klassensystem selbst ist ein neues Konzept und als solches sicherlich noch nicht ausgereift. Bei seiner Verwendung kann sich herausstellen, dass einzelne Klassifikationen weitaus hilfreicher sind als andere, dass nützliche Klassifikationen fehlen oder dass aus der Zuordnung von Ähnlichkeitsmaßen Schlüsse gezogen werden, die aus verschiedenen Gründen nicht zielführend im Bezug auf die Anwendung sind. Insbesondere die letzten Klassifikationen sind ein Indiz für die genannten Punkte. Weiterhin kann es für einige Charakteristika sinnvoll sein, aus ihnen nicht nur diskrete Klassen abzuleiten, sondern sie zu quantifizieren, zum Beispiel die Empfindlichkeit auf Ausreißer (Klassifikation AR). In der Folge bedarf die vorgeschlagene Systematik gründlicher Prüfung auf Praxistauglichkeit und Nützlichkeit und sollte gegebenenfalls angepasst und erweitert werden. Dabei muss stets die Korrektheit und Generizität der Klassendefinitionen sowie der Zusammenhang der Klassifikationen untereinander beachtet werden.

Die Ergebnisse der vorliegenden Arbeit schöpfen folglich nicht alle Möglichkeiten der Thematik aus, sondern sind ein erster Schritt, sich der Problematik anzunehmen. Insofern stellt jene vor dem Hintergrund der derzeitigen Situation der fehlenden Übersicht über die Vielzahl an Techniken für den Vergleich von spatiotemporalen Trajektorien ein wertvolles Forschungsergebnis für den Fachbereich dar.

Jahr	Vergleichstechnik	Ä'maß	Publikation(en)	Autor(en)
-	Closest-Pair-Distance	d_{CPD}	-	-
-	Aggregate über synchrone Glieder	d_{AI}	[AFS93, FRM94]	-
-	Euklidische Distanz	d_{E}	-	-
-	Aggregate über PTD	d_{AP}	-	-
-	Fréchet-Distanz	$d_{Fréchet}$	[AG92]	Maurice Fréchet
1914	Hausdorff-Distanz	$d_{Hausdorff}$	[Hau14]	Felix Hausdorff
1994	Dynamic-Time-Warping	d_{DTW}	[BC94]	Berndt & Clifford
1995	Envelope-Technik	-	[LS95]	Agrawal e.a.
2002	LCSS	d_{LCSS}	[VKG02]	Vlachos e.a.
2002	LCSS, Sigmoidfkt.	d_{LCSS_SM}	[VGK02]	Vlachos e.a.
2003	Shape-Based-Distance	d_{SBD}	[YAS03]	Yanagisawa e.a.
2003	Area-Based-Distance	d_{ABD}	[NB03]	Needham & Boyle
2004	AAL-Warping	d_{AAL}	[VGD04]	Vlachos e.a.
2004	HMM	-	[Por04]	Porikli
2005	Edit-Distance-on-Real-Sequence	d_{EDR}	[CÖO05, Che05]	Chen
2005	Edit-Distance-with-Real-Penalty	d_{ERP}	[CN04b] [Che05]	Chen
2005	RN-Technik	s_{RN}, d_{RN}	[HKL05, HKL06]	Hwang e.a.
2005	Grid-basierter Ansatz	d_{Grid}	[LS05]	Lin & Su
2006	Modifizierte Hausdorff-Distanz	d_{MOHD}	[AMP06]	Atev e.a.
2006	Piciarelli-Foresti-Distanz	d_{PF}	[PF06]	Piciarelli & Foresti
2006	Threshold-Distanz	d_{TQuEST}	[AKK+06]	Aßfalg e.a.
2007	DISSIM	d_{DISSIM}	[FGT07]	Frentzos e.a.
2007	Swale	s_{Swale}	[MP07]	Morse & Patel
2007	Spatial-Assembling-Distance	d_{SpADe}	[CNOT07]	Nascimento e.a.
2007	HU-Distanz	-	[HXF+07]	Hu e.a.
2007	PDM	-	[RMJ07]	Roduit e.a.
2008	Graph-basierte Distanz	d_{Graph}	[TPN+09]	Tiakas e.a.
2010	Maximal-Semantic-Trajectory-Patterns	s_{MSTP}	[YLL+10, CLMP14]	Ying e.a.
2012	Sequence-Pattern-Mining	d_{SPM}	[YCW+12]	Yang e.a.

Tabelle 7.1: Chronologische Übersicht über Vergleichstechniken

Ä'maß	MR	ZS	ZI	LN	AK	EL	LE	ZE
d_{CPD}	0	0	0	0	E	E	0	0
d_{AI_min}	0	I	0	L	E	0	0	P
d_{AI_max}	0	I	0	L	E	0	0	P
d_{AI_sum}	0	I	0	L	A	0	L	P
d_{AI_mean}	0	I	0	L	A	0	0	P
d_{AI_RMS}	0	I	0	L	A	0	0	P
d_{AI_median}	0	I	0	L	E(M)	0	0	P
d_{E}	0	I	0	L	A	0	L	P
d_{AP_mean}	0	0	0	0	A	E	0	0
$d_{Hausdorff}$	0	0	0	0	E	E	0	0
d_{MOHD}	0	0	0	0	E	E	0	P
$d_{Fréchet}$	0	0	0	0	E	E	0	P
d_{DTW}	0	0	0(Ä)	0	A	E	L	P
d_{LCSS_1}	0	0	0(Ä)	0	M	E	0	P
d_{LCSS_2}	0	0	0(Ä)	0	M	E	0	P
$d_{LCSS\_SM\_1}$	0	0	0(Ä)	0	M	E	0	P
$d_{LCSS\_SM\_2}$	0	0	0(Ä)	0	M	E	0	P
d_{EDR}	0	0	0	0	A	E	L	P
d_{ERP}	0	0	0	0	A	E	L	P
s_{Swale}	0	0	0	0	A	E	L	P
d_{PF}	0	I	Ä	L	A	E	0	P
d_{SBD_TS}	(0)	0(I)	0(Ä)	T	A	0(E)	0	Z
d_{SBD_S}	(0)	0	0	S	A	0(E)	0	Z
d_{ABD}	0	0(I)	0	0	A	-	L	P
d_{DISSIM}	0	R(I)	0	T	A	-	L	Z
d_{SPM}	(M)	I	0	L	A	0	0	P
d_{AAL}	M	0	0	0	A	-	L	P
d_{SpADe}	M	0	Ä	0	M	-	L	-
d_{TQuEST}	(M)	0	0	0	M	-	0	Z
d_{RN_T}	0	0	0	0	(E)	E	0	Z
d_{RN_S}	0	(I)	0	0	M	E	0	Z
d_{RN_TS}	0	(I)	0	0	M	E	0	Z
d_{Graph}	0	I	0	L	A	0	0	P
d_{Grid}	0	0	0	0	A	E	0	0

Tabelle 7.2: Ähnlichkeitsmaße und deren Charakteristika (1/2)

Ä'maß	MD	PRM	MTR	KMP	SI	AR
d_{CPD}	1	0	0	$\mathcal{O}(n^2)$	I	M
d_{AI_min}	1	0	0	$\mathcal{O}(n)$	0	M
d_{AI_max}	1	0	M	$\mathcal{O}(n)$	0	M
d_{AI_sum}	1	0	M	$\mathcal{O}(n)$	0	I
d_{AI_mean}	1	0	M	$\mathcal{O}(n)$	0	I
d_{AI_RMS}	1	0	M	$\mathcal{O}(n)$	0	I
d_{AI_median}	1	0	0	$\mathcal{O}(n)$	0	0
d_{E}	1	0	M	$\mathcal{O}(n)$	0	I
d_{AP_mean}	1	0	M	$\mathcal{O}(n^2)$	I	I
$d_{Hausdorff}$	1	0	M	$\mathcal{O}(n^2)$	I	I
d_{MOHD}	1	WV	0(M)	$\mathcal{O}(nw)$	I	0
$d_{Fréchet}$	1	0	M	$\mathcal{O}(n^2)$	I	I
d_{DTW}	1	(W)	S	$\mathcal{O}(n^2)$	0	I
d_{LCSS_1}	0	WP	0	$\mathcal{O}(n^2)$	0	0
d_{LCSS_2}	0	WP	0	$\mathcal{O}((2n)^3\delta^3)$	0	0
$d_{LCSS\_SM\_1}$	0	WV	0	$\mathcal{O}(n^2)$	0	0
$d_{LCSS\_SM\_2}$	0	WV	0	$\mathcal{O}((2n)^3\delta^3)$	0	0
d_{EDR}	0	P	0	$\mathcal{O}(n^2)$	0	0
d_{ERP}	1	0	M	$\mathcal{O}(n^2)$	0	I
s_{Swale}	0	PV	0	$\mathcal{O}(n^2)$	0	0
d_{PF}	1	W	P	$\mathcal{O}(\delta n)$	0	(I)
d_{SBD_TS}	1	0(V)	M	$(\mathcal{O}(n))$	I	(I)
d_{SBD_S}	1	0(V)	M	$(\mathcal{O}(n))$	I	I
d_{ABD}	2	0	(M)	-	I	I
d_{DISSIM}	2	0	M	-	I	I
d_{SPM}	0	V	0	$\mathcal{O}(n)$	0	I
d_{AAL}	0	0	0	$\mathcal{O}(n^2)$	0	I
d_{SpADe}	0	V	0	$\mathcal{O}(n^2)$	-	0
d_{TQuEST}	0	A	0	$\mathcal{O}(N_q N_k \log(N_p))$	I	M
d_{RN_T}	1	A	0	$(\mathcal{O}(n))$	-	-
d_{RN_S}	1	A	0	$(\mathcal{O}(n))$	-	-
d_{RN_TS}	0	A	0	$(\mathcal{O}(n))$	-	-
d_{Graph}	1	0	M	$\mathcal{O}(n)$	-	-
d_{Grid}	1	0	M	$\mathcal{O}(ni)$	-	I

Tabelle 7.3: Ähnlichkeitsmaße und deren Charakteristika (2/2)

Abbildungsverzeichnis

Tabellenverzeichnis

Definitionsverzeichnis

Abkürzungsverzeichnis

AAL	Angle-/Arc-Length
ABD	Area-Based-Distance
APCA	Adaptive-Piecewise-Constant-Approximation
BCT	Best-Connected-Trajectory
CHEB	Chebyshev-Polynomials
CAI	Common-Appeareance-Interval
CPD	Closest-Pair-Distance
CPS	Common-Pattern-Sets
DFT	Diskrete Fourier-Transformation
DWT	Diskrete Wavelet-Transformation
DTW	Dynamic-Time-Warping
EDR	Edit-Distance-on-Real-Sequence
ERP	Edit-Distance-with-Real-Penalty
FTSE	Fast-Time-Series-Evaluation
FTW	Fast-Search-Method-for-Dynamic-Time-Warping
HMM	Hidden-Markov-Model
IPLA	Indexable-Piecewise-Linear-Approximation
LCSS	Longest-Common-Subsequence
LBSN	Location-Based-Social-Network
MBFD	Minimum-Backward-Fréchet-Distance
MOHD	Modifizierte Hausdorff-Distanz
MSTP	Maximal-Semantic-Trajectory-Pattern
MTP	Maximal-Trajectory-Pattern
OWD	One-Way-Distance
PAA	Piecewise-Aggregate-Approximation
PCA	Piecewise-Constant-Approximation
PDM	Point-Distribution-Model
PTD	Positions-Trajektorien-Distanz
PLA	Piecewise-Linear-Approximation
PoI	Points-of-Interest

RoI	Region-of-Interest
RN	Road-Network
SpADe	Spatial-Assembling-Distance
Swale	Sequence-Weighted-Alignment-Model
SAX	Symbolic-Aggregate-Approximation
SBD	Shape-Based-Distance
SPM	Sequence-Pattern-Mining
SVD	Single-Value-Decomposition
TCT	Threshold-Crossing-Time-Intervals
ToI	Times-of-Interest
TQuEST	Threshold-Queries-Similarity-Search

Literaturverzeichnis

[AFS93] Agrawal, Rakesh, Christos Faloutsos und Arun Swami: *Efficient similarity search in sequence databases.* In: *International Conference on Foundations of Data Organization and Algorithms*, Seiten 69–84. Springer, 1993.

[AG92] Alt, Helmut und Michael Godau: *Measuring the Resemblance of Polygonal Curves.* In: *Proceedings of the Eighth Annual Symposium on Computational Geometry*, SCG '92, Seiten 102–109, New York, NY, USA, 1992. ACM.

[AKK+06] Assfalg, Johannes, Hans-Peter Kriegel, Peer Kröger, Peter Kunath, Alexey Pryakhin und Matthias Renz: *Similarity search on time series based on threshold queries.* In: *International Conference on Extending Database Technology*, Seiten 276–294. Springer, 2006.

[AMP06] Atev, Stefan, Osama Masoud und Nikolaos Papanikolopoulos: *Learning Traffic Patterns at Intersections by Spectral Clustering of Motion Trajectories.* In: *IROS*, Seiten 4851–4856, 2006.

[BC57] Bray, J Roger und John T Curtis: *An ordination of the upland forest communities of southern Wisconsin.* Ecological monographs, 27(4):325–349, 1957.

[BC94] Berndt, Donald J und James Clifford: *Using Dynamic Time Warping to Find Patterns in Time Series.* In: *KDD workshop*, Band 10, Seiten 359–370. Seattle, WA, 1994.

[BER+03] Black, James, Tim Ellis, Paul Rosin et al.: *A novel method for video tracking performance evaluation.* Proceedings of the IEEE International-Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS 03), Seiten 125–132, 2003.

[BI14] Backurs, Arturs und Piotr Indyk: *Edit Distance Cannot Be Computed in Strongly Subquadratic Time (unless SETH is false).* CoRR, abs/1412.0348, 2014.

[BKA09] Bogorny, Vania, Bart Kuijpers und Luis Otavio Alvares: *ST-DMQL: a semantic trajectory data mining query language.* International Journal of Geographical Information Science, 23(10):1245–1276, 2009.

[Bri14] Bringmann, Karl: *Why walking the dog takes time: Frechet distance has no strongly subquadratic algorithms unless SETH fails.* CoRR, abs/1404.1448, 2014.

[BYÖ97] Bozkaya, Tolga, Nasser Yazdani und Meral Özsoyoğlu: *Matching and indexing sequences of different lengths.* In: *Proceedings of the sixth international conference on Information and knowledge management*, Seiten 128–135. ACM, 1997.

[CCL+07] Chen, Qiuxia, Lei Chen, Xiang Lian, Yunhao Liu und Jeffrey Xu Yu: *Indexable PLA for efficient similarity search.* In: *Proceedings of the 33rd international conference on Very large data bases*, Seiten 435–446. VLDB Endowment, 2007.

[CF99] Chan, Kin-Pong und Ada Wai-Chee Fu: *Efficient time series matching by wavelets.* In: *Data Engineering, 1999. Proceedings., 15th International Conference on*, Seiten 126–133. IEEE, 1999.

[Che05] Chen, Lei: *Similarity search over time series and trajectory data.* Doktorarbeit, University of Waterloo, 2005.

[CLMP14] Chen, Xihui, Ruipeng Lu, Xiaoxing Ma und Jun Pang: *Measuring user similarity with trajectory patterns: Principles and new metrics.* In: *Asia-Pacific Web Conference*, Seiten 437–448. Springer, 2014.

[CM99] Cohen, Isaac und Gerard Medioni: *Detecting and tracking moving objects for video surveillance.* In: *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, Band 2. IEEE, 1999.

[CN04a] Cai, Yuhan und Raymond Ng: *Indexing spatio-temporal trajectories with Chebyshev polynomials.* In: *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, Seiten 599–610. ACM, 2004.

[CN04b] Chen, Lei und Raymond Ng: *On the marriage of lp-norms and edit distance.* In: *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, Seiten 792–803. VLDB Endowment, 2004.

[CNOT07] Chen, Yueguo, Mario A Nascimento, Beng Chin Ooi und Anthony KH Tung: *Spade: On shape-based pattern detection in streaming time series.* In: *2007 IEEE 23rd International Conference on Data Engineering*, Seiten 786–795. IEEE, 2007.

[CÖO05] Chen, Lei, M Tamer Özsu und Vincent Oria: *Robust and fast similarity search for moving object trajectories.* In: *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, Seiten 491–502. ACM, 2005.

[CPX13] Chen, Xihui, Jun Pang und Ran Xue: *Constructing and comparing user mobility profiles for location-based services.* In: *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, Seiten 261–266. ACM, 2013.

[CPX14] Chen, Xihui, Jun Pang und Ran Xue: *Constructing and comparing user mobility profiles.* 8(4):21, 2014.

[CSZ+10] Chen, Zaiben, Heng Tao Shen, Xiaofang Zhou, Yu Zheng und Xing Xie: *Searching trajectories by locations: an efficiency study.* In: *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, Seiten 255–266. ACM, 2010.

[DN05] D'Auria, Dino Pedreschi Margherita und Mirco Nanni: *Prediction time-focused density-based clustering of trajectories of moving objects.* In: *SIGMOD'04: Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, 2005.

[DTS+08] Ding, Hui, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang und Eamonn Keogh: *Querying and mining of time series data: experimen-*

tal comparison of representations and distance measures. 1(2):1542–1552, 2008.

[EM94] Eiter, Thomas und Heikki Mannila: *Computing discrete Fréchet distance.* Technischer Bericht, Citeseer, 1994.

[FGPT07] Frentzos, Elias, Kostas Gratsias, Nikos Pelekis und Yannis Theodoridis: *Algorithms for nearest neighbor search on moving object trajectories.* Geoinformatica, 11(2):159–193, 2007.

[FGT07] Frentzos, Elias, Kostas Gratsias und Yannis Theodoridis: *Index-based most similar trajectory search.* In: *2007 IEEE 23rd International Conference on Data Engineering*, Seiten 816–825. IEEE, 2007.

[FRM94] Faloutsos, Christos, Mudumbai Ranganathan und Yannis Manolopoulos: *Fast subsequence matching in time-series databases*, Band 23. ACM, 1994.

[GK95] Goldin, Dina Q und Paris C Kanellakis: *On similarity queries for time-series data: constraint specification and implementation.* In: *International Conference on Principles and Practice of Constraint Programming*, Seiten 137–153. Springer, 1995.

[GNPP07] Giannotti, Fosca, Mirco Nanni, Fabio Pinelli und Dino Pedreschi: *Trajectory pattern mining.* In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, Seiten 330–339. ACM, 2007.

[GTW+10] Ghica, Oliviu, Goce Trajcevski, Ouri Wolfson, Ugo Buy, Peter Scheuermann, Fan Zhou und Dennis Vaccaro: *Trajectory data reduction in wireless sensor networks.* 1(1), 2010.

[Hau] Hauser, Rainer: *Zahlen ist Messen in der nullten Dimension. Oder: Was macht man mit halben Koordinatenachsen?*

[Hau14] Hausdorff, Felix: *Grundzüge der Mengenlehre.* Leipzig, 1914.

[HBC+05] Hampapur, Arun, Lisa Brown, Jonathan Connell, Ahmet Ekin, Norman Haas, Max Lu, Hans Merkl und Sharath Pankanti: *Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking.* IEEE Signal Processing Magazine, 22(2):38–51, 2005.

[HG97] Heckbert, Paul S und Michael Garland: *Survey of polygonal surface simplification algorithms.* Technischer Bericht, DTIC Document, 1997.

[HKL05] Hwang, Jung-Rae, Hye-Young Kang und Ki-Joune Li: *Spatio-temporal similarity analysis between trajectories on road networks.* In: *International Conference on Conceptual Modeling*, Seiten 280–289. Springer, 2005.

[HKL06] Hwang, Jung-Rae, Hye-Young Kang und Ki-Joune Li: *Searching for similar trajectories on road networks using spatio-temporal similarity.* In: *East European Conference on Advances in Databases and Information Systems*, Seiten 282–295. Springer, 2006.

[HKR93] Huttenlocher, Daniel P., Gregory A. Klanderman und William J Rucklidge: *Comparing images using the Hausdorff distance.* IEEE Transactions on pattern analysis and machine intelligence, 15(9):850–863, 1993.

[HXF+07] Hu, Weiming, Dan Xie, Zhouyu Fu, Wenrong Zeng und Steve Maybank: *Semantic-based surveillance video retrieval.* IEEE Transactions on image processing, 16(4):1168–1181, 2007.

[JWG00] Jacobs, David W, Daphna Weinshall und Yoram Gdalyahu: *Classification with nonmetric distances: Image retrieval and class representation.* 22(6):583–600, 2000.

[KCPM01] Keogh, Eamonn, Kaushik Chakrabarti, Michael Pazzani und Sharad Mehrotra: *Dimensionality reduction for fast similarity search in large time series databases.* Knowledge and information Systems, 3(3):263–286, 2001.

[Keo06] Keogh, Eamonn: *A decade of progress in indexing and mining large time series databases.* In: *Proceedings of the 32nd international conference on Very large data bases*, Seiten 1268–1268. VLDB Endowment, 2006.

[KGP01] Kalpakis, Konstantinos, Dhiral Gada und Vasundhara Puttagunta: *Distance measures for effective clustering of ARIMA time-series.* In: *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, Seiten 273–280. IEEE, 2001.

[KP99] Keogh, Eamonn J und Michael J Pazzani: *Relevance feedback retrieval of time series data.* In: *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, Seiten 183–190. ACM, 1999.

[KPC01] Kim, Sang-Wook, Sanghyun Park und Wesley W Chu: *An index-based approach for similarity search supporting time warping in large sequence databases.* In: *Data Engineering, 2001. Proceedings. 17th International Conference on*, Seiten 607–614. IEEE, 2001.

[KR05] Keogh, Eamonn und Chotirat Ann Ratanamahatana: *Exact indexing of dynamic time warping.* 7(3):358–386, 2005.

[KWX+06] Keogh, Eamonn, Li Wei, Xiaopeng Xi, Sang-Hee Lee und Michail Vlachos: *LB_Keogh supports exact indexing of shapes under rotation invariance with arbitrary representations and distance measures.* In: *Proceedings of the 32nd international conference on Very large data bases*, Seiten 882–893. VLDB Endowment, 2006.

[Leb02] Lebesgue, Henri: *Intégrale, longueur, aire.* Annali di Matematica Pura ed Applicata (1898-1922), 7(1):231–359, 1902.

[Lev66] Levenshtein, Vladimir I: *Binary codes capable of correcting deletions, insertions and reversals.* In: *Soviet physics doklady*, Band 10, Seite 707, 1966.

[LHW07] Lee, Jae-Gil, Jiawei Han und Kyu-Young Whang: *Trajectory clustering: a partition-and-group framework.* In: *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, Seiten 593–604. ACM, 2007.

[LKWL07] Lin, Jessica, Eamonn Keogh, Li Wei und Stefano Lonardi: *Experiencing SAX: a novel symbolic representation of time series.* Data Mining and knowledge discovery, 15(2):107–144, 2007.

[LRH11] Lawson, Catherine T, SS Ravi und Jeong-Hyon Hwang: *Compression and Mining of GPS Trace Data: New Techniques and Applications.* Final Report: Region II University Transportation Research Center, University at Albany-SUNY, 2011.

[LS95] Lin, Rake& Agrawal King-lp und Harpreet S Sawhney Kyuseok Shim: *Fast similarity search in the presence of noise, scaling, and translation in time-series databases.* In: *Proceeding of the 21th International Conference on Very Large Data Bases*, Seiten 490–501. Citeseer, 1995.

[LS05] Lin, Bin und Jianwen Su: *Shapes based trajectory queries for moving objects.* In: *Proceedings of the 13th annual ACM international workshop on Geographic information systems*, Seiten 21–30. ACM, 2005.

[Mai78] Maier, David: *The Complexity of Some Problems on Subsequences and Supersequences.* J. ACM, 25(2):322–336, April 1978.

[MdB02] Meratnia, Nirvana und Rolf A de By: *Aggregation and comparison of trajectories.* In: *Proceedings of the 10th ACM international symposium on Advances in geographic information systems*, Seiten 49–54. ACM, 2002.

[MP07] Morse, Michael D und Jignesh M Patel: *An efficient and accurate method for evaluating time series similarity.* In: *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, Seiten 569–580. ACM, 2007.

[MR04] Meratnia, Nirvana und A Rolf: *Spatiotemporal compression techniques for moving point objects.* In: *International Conference on Extending Database Technology*, Seiten 765–782. Springer, 2004.

[MT08] Morris, Brendan Tran und Mohan Manubhai Trivedi: *A survey of vision-based trajectory learning and analysis for surveillance.* IEEE transactions on circuits and systems for video technology, 18(8):1114–1127, 2008.

[MT09] Morris, Brendan und Mohan Trivedi: *Learning trajectory patterns by clustering: Experimental studies and comparative evaluation.* In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, Seiten 312–319. IEEE, 2009.

[MWH02] Moon, Yang-Sae, Kyu-Young Whang und Wook-Shin Han: *General match: a subsequence matching method in time-series databases based on generalized windows.* In: *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, Seiten 382–393. ACM, 2002.

[NB03] Needham, Chris J und Roger D Boyle: *Performance evaluation metrics and statistics for positional tracker evaluation.* In: *International Conference on Computer Vision Systems*, Seiten 278–289. Springer, 2003.

[OTC09] Oda, Junichi, Ruck Thawonmas und Kuan-Ta Chen: *Comparison of user trajectories based on coordinate data and state transitions.* In: *Intelligent Information Hiding and Multimedia Signal Processing, 2009. IIH-MSP'09. Fifth International Conference on*, Seiten 1134–1137. IEEE, 2009.

[PF06] Piciarelli, Claudio und Gian Luca Foresti: *On-line trajectory clustering for anomalous events detection.* 27(15):1835–1842, 2006.

[Por04] Porikli, Fatih: *Trajectory distance metric using hidden markov model based representation.* In: *IEEE European Conference on Computer Vision, PETS Workshop*, Band 3, Seite 153, 2004.

[RMJ07] Roduit, Pierre, Alcherio Martinoli und Jacques Jacot: *A quantitative method for comparing trajectories of mobile robots using point distribution models.* In: *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Seiten 2441–2448. IEEE, 2007.

[RSH+11] Rhee, Injong, Minsu Shin, Seongik Hong, Kyunghan Lee, Seong Joon Kim und Song Chong: *On the levy-walk nature of human mobility.* IEEE/ACM transactions on networking (TON), 19(3):630–643, 2011.

[SAF94] Scassellati, Brian M, Sophoclis Alexopoulos und Myron D Flickner: *Retrieving images by 2D shape: a comparison of computation methods with human perceptual judgments.* In: *IS&T/SPIE 1994 International Symposium on Electronic Imaging: Science and Technology*, Seiten 2–14. International Society for Optics and Photonics, 1994.

[SUS95] Slater, Mel, Martin Usoh und Anthony Steed: *Taking steps: the influence of a walking technique on presence in virtual reality.* ACM Transactions on Computer-Human Interaction (TOCHI), 2(3):201–219, 1995.

[SYF05] Sakurai, Yasushi, Masatoshi Yoshikawa und Christos Faloutsos: *FTW: fast similarity search under the time warping distance.* In: *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, Seiten 326–337. ACM, 2005.

[TMM+11] Terziman, Léo, Maud Marchal, Franck Multon, Bruno Arnaldi und Anatole Lécuyer: *Comparing virtual trajectories made in slalom using walking-in-place and joystick techniques.* In: *EuroVR/EGVE Joint Virtual Reality Conference.* Eurographics, 2011.

[TPN+09] Tiakas, Eleftherios, AN Papadopoulos, Alexandros Nanopoulos, Yannis Manolopoulos, Dragan Stojanovic und Slobodanka Djordjevic-Kajan: *Searching for similar trajectories in spatial networks.* 82(5):772–788, 2009.

[VCB+88] Vidal, Enrique, Francisco Casacuberta, Jose M Benedi, Maria J Lloret und Hector Rulot: *On the verification of triangle inequality by dynamic time-warping dissimilarity measures.* Speech communication, 7(1):67–79, 1988.

[VGD04] Vlachos, Michail, Dimitrios Gunopulos und Gautam Das: *Rotation invariant distance measures for trajectories.* In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, Seiten 707–712. ACM, 2004.

[VGK02] Vlachos, Michail, Dimitrios Gunopulos und George Kollios: *Robust similarity measures for mobile object trajectories.* In: *Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on*, Seiten 721–726. IEEE, 2002.

[Vit08] Vitali, Giuseppe: *Sui gruppi di puni e sulle funzioni di variabili reali.* 1908.

[VKG02] Vlachos, Michail, George Kollios und Dimitrios Gunopulos: *Discovering similar multidimensional trajectories.* In: *Data Engineering, 2002. Proceedings. 18th International Conference on*, Seiten 673–684. IEEE, 2002.

[WMD+13] Wang, Xiaoyue, Abdullah Mueen, Hui Ding, Goce Trajcevski, Peter Scheuermann und Eamonn Keogh: *Experimental comparison of representation methods and distance measures for time series data.* 26(2):275–309, 2013.

[Wol02] Wolfson, Ouri: *Moving objects information management: The database challenge.* In: *International Workshop on Next Generation Information Technologies and Systems*, Seiten 75–89. Springer, 2002.

[XX15] Xinghua Xia, Dafang Yang, Fangjun Luan: *An Efficient DTW Matching for On-line Signature Verification.* 6th International Conference on Electronics, Mechanics, Culture and Medicine, 2015.

[YAS03] Yanagisawa, Yutaka, Jun-ichi Akahani und Tetsuji Satoh: *Shape-based similarity query for trajectory of mobile objects.* In: *International Conference on Mobile Data Management*, Seiten 63–77. Springer, 2003.

[YCW+12] Yang, Yuanfeng, Z Cui, J Wu, G Zhang und X Xian: *Trajectory analysis using spectral clustering and sequence pattern mining.* 8(6):2637–2645, 2012.

[YLL+10] Ying, Josh Jia-Ching, Eric Hsueh-Chan Lu, Wang-Chien Lee, Tz-Chiao Weng und Vincent S Tseng: *Mining user similarity from semantic trajectories.* In: *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks*, Seiten 19–26. ACM, 2010.

[ZCZC09] Zhang, Chengcui, Xin Chen, Liping Zhou und Wei-Bang Chen: *Semantic retrieval of events from indoor surveillance video databases.* Pattern Recognition Letters, 30(12):1067–1076, 2009.

[ZHT06] Zhang, Zhang, Kaiqi Huang und Tieniu Tan: *Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes.* In: *18th International Conference on Pattern Recognition (ICPR'06)*, Band 3, Seiten 1135–1138. IEEE, 2006.

[ZZ11] Zheng, Yu und Xiaofang Zhou: *Computing with spatial trajectories.* Springer Science & Business Media, 2011.